AD-A115 510 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 1/2
DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS VIA COMMAND G--ETC(U)
DEC 81 R M FLOYD
UNCLASSIFIED AFIT/GE/EE/81-20-VOL-1 NL

1 of 3
AD
A115 510

AD A115510

DTIC FILE COPY

DTIC
ELECTE
JUN 14 1982
S D

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY (ATC)

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

82 06 14 079

AFIT/GE/EE/81-20

DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS VIA COMMAND GENERATOR TRACKER (CGT) SYNTHESIS METHODS

THESIS VOLUME I

AFIT/GE/EE/81-20-VOL-1

Richard M. Floyd
Capt USAF

DTIC
ELECTE
JUN 14 1982
D
H

Approved for public release; distribution unlimited

AFIT/GE/EE/81-20

DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS VIA COMMAND GENERATOR TRACKER (CGT) SYNTHESIS METHODS

THESIS VOLUME I

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by
Richard M. Floyd, B.S.
Capt USAF
Graduate Electrical Engineering
December 1981

Accession For
NTIS GRA&I
DTIC TAB
Unannounced
Justification
By
Distribution/
Availability Codes
Dist | Avail and/or Special
A

DTIC
COPY
INSPECTED
2

Approved for public release; distribution unlimited

Preface

As a flight control engineer in the Flight Dynamics Laboratory of the Air Force Wright Aeronautical Laboratories, I had the opportunity to work with advanced aircraft control objectives and designs. I was impressed with the difficulty of achieving ultimate aircraft control behavior when employing conventional design techniques derived from designs for single-input single-output (SISO) systems. There is a substantial gulf between techniques and design methodologies of the so-called modern control theory and the designs that are actually pursued by practicing flight control engineers. This gulf is becoming still broader in the realm of controller design for sampled-data implementation.

I have held an interest for several years in optimal model-following design as a method for achieving advanced flight control systems for sampled-data implementations. The Command Generator Tracker control system considered in this study is a new development of the model-following design method, and offers several advantages over earlier such designs.

I wish to thank my thesis advisor, Dr. Peter S. Maybeck of the Air Force Institute of Technology, for his interest in this study. His consistent concern for thorough and accurate research and reporting is the model for my own efforts in this and future studies I may undertake.

Contents

Page

Preface ii

List of Figures vi

List of Tables ix

Abstract x

I. Introduction 1

- 1.1 Background 1
- 1.2 Problem 4
- 1.3 Sequence of Presentation 4

II. Model-Following Control 7

- 2.1 Introduction 7
- 2.2 Model-Following Control in Aircraft Design . 8
- 2.3 Types of Model-Following Controllers 9
  - 2.3.1 Implicit Model-Following Controller . 10
  - 2.3.2 Explicit Model-Following Controller . 13
- 2.4 Comparison of Implicit and Explicit Model-Following 16
- 2.5 Introduction to the CGT/PI/KF Controller . . 19

III. CGT/PI/KF Theoretical Development 22

- 3.1 Overview of the Theory 22
- 3.2 System Models 26
  - 3.2.1 Design Model 27
  - 3.2.2 Truth Model 32
- 3.3 Open-Loop Command Generator Tracker (CGT) . . 35
  - 3.3.1 Command Model 35
  - 3.3.2 Ideal Trajectory 36
  - 3.3.3 Open-Loop CGT 38
- 3.4 Optimal Regulator/PI Controller 44
  - 3.4.1 Control Difference PI Controller . . 47
  - 3.4.2 Continuous-Time Cost 49
  - 3.4.3 Achieving Integral Control 55
- 3.5 Closed-Loop CGT/PI 58
  - 3.5.1 Perturbation Regulator 58
  - 3.5.2 Achieving Integral Control 60
  - 3.5.3 CGT/PI Control Law 63

Page

3.6 Kalman Filter 64
 3.6.1 Design Model 66
 3.6.2 Steady-State Kalman Filter 66
 3.6.3 State Estimates 68

3.7 CGT/PI/KF Control Law 69

IV. CGT/PI/KF Design Evaluation 71

4.1 Introduction 71
4.2 PI Regulator Evaluation 72
4.3 CGT or CGT/PI Evaluation 74
4.4 Kalman Filter Evaluation 75

V. CGT/PI/KF Design Computer Program 82

5.1 Introduction 82
5.2 Program Operating Principles and Organization 83
 5.2.1 Interactive Execution 86
 5.2.2 Array Allocation 87
 5.2.3 Entry of Dynamics Models 88
 5.2.4 Design Paths 89
 5.2.5 Input Prompts 89
 5.2.6 Program Output 90
 5.2.7 Preservation of Design Information . 91
 5.2.8 Error Checking 92
5.3 Program Usage 93

VI. CGT Design Results 95

6.1 Introduction 95
 6.1.1 Design Examples 95
 6.1.2 PI Regulator Design for CGT/PI Application 99
 6.1.3 Determining Command Models 102
 6.1.4 Need for Complete Evaluation Software 103
6.2 Using the CGTPIF Design Program 104
 6.2.1 Introduction 104
 6.2.2 Operating Considerations 104
 6.2.3 Interpreting Plots to the Terminal . 108
6.3 Simple Design Example 109
 6.3.1 Design Model 109
 6.3.2 Truth Model 111
 6.3.3 Command Models 112
  6.3.3.1 First-Order Command Model CM(O1) 112
  6.3.3.2 Second-Order Command Model CM(O2C), CM(O2C)´. . . 113

Page

6.3.4 Design for CM(O1) 114
6.3.5 Design for CM(O2C) and CM(O2C)$'$ . . . 120
6.3.6 Discussion of Results for Simple Design Example 129

6.4 Aircraft Flight Control Design 131
6.4.1 Introduction 131
6.4.2 AFTI/F-16 Truth Model 134
6.4.3 AFTI/F-16 Model Simplifications: Design Models 145
6.4.3.1 Model AFTI(S3,A2,G3) 146
6.4.3.2 Model AFTI(S3,A2). 150
6.4.3.3 Model AFTI(S3) 151
6.4.3.3.1 Truth Model TM(S3)$^{+}$ 152
6.4.3.3.2 Truth Model TM(S3)$^{-}$ 153
6.4.3.3.3 Truth Model TM(S3)$'$. 153
6.4.4 Model Output Equations 154
6.4.4.1 Output Equations for Pitch-Pointing Control . . . 154
6.4.4.2 Output Equations for Pitch Rate Control 155
6.4.5 Command Models 155
6.4.5.1 Command Model for Pitch-Pointing Control 155
6.4.5.2 Command Model for Pitch Rate Control 159
6.4.6 CGT/PI Pitch-Pointing Controller Design 160
6.4.6.1 AFTI(S3) Pitch-Pointing CGT/PI 160
6.4.6.2 AFTI(S3,A2) Pitch-Pointing CGT/PI 172
6.4.6.3 AFTI(S3,A2,G3) Pitch-Pointing CGT/PI 176
6.4.7 CGT/PI Pitch Rate Controller Design . 179
6.4.7.1 AFTI(S3) Pitch Rate CGT/PI . 180
6.4.7.2 AFTI(S3,A2) Pitch Rate CGT/PI 183
6.4.8 Kalman Filter Design 191
6.4.9 Discussion of Results for Flight Control Design Example. 201

VII. Conclusions and Recommendations for Future Research 203

7.1 Conclusions 203
7.2 Recommendations for Future Research 205

Bibliography 208

Vita 211

List of Figures

Figure Page

3-1. CGT/PI/KF General Block Diagram 24

3-2. Open-Loop Command Generator Tracker 45

5-1. CGTPIF General Flowchart 85

6-1. Open-Loop CGT for CM(O1), (No Disturbances, Nominal Truth Model) 116

6-2. PI Regulator Response (Simple Design Example) 118

6-3. CGT/PI for CM(O1), (No Disturbances, Nominal Truth Model) 119

6-4. Open-Loop CGT for CM(O2C), (No Disturbances, Nominal Truth Model) 122

6-5. Open-Loop CGT for CM(O2C), (Disturbances, Nominal Truth Model) 123

6-6. Open-Loop CGT for CM(O2C), (No Disturbances, Alternate Truth Model) 124

6-7. CGT/PI for CM(O2C), (No Disturbances, Nominal Truth Model) 126

6-8. CGT/PI for CM(O2C), (Disturbances, Nominal Truth Model) 127

6-9. CGT/PI for CM(O2C), (No Disturbances, Alternate Truth Model) 128

6-10. AFTI(S3) PI Regulator--No Weight on Pitch Rate 162

6-11. AFTI(S3) Pitch-Pointing CGT/PI--No Weight on Pitch Rate 163

6-12. AFTI(S3) PI Regulator--Weight on Pitch Rate 164

6-13. AFTI(S3) Pitch-Pointing CGT/PI--Weight on Pitch Rate 165

Figure Page

6-14. AFTI(S3) CGT/PI Control Surface Deflections 167

6-15. AFTI(S3) Pitch-Pointing CGT/PI WRT TM(S3)$^+$ 169

6-16. AFTI(S3) Pitch-Pointing CGT/PI WRT TM(S3)$^-$ 170

6-17. AFTI(S3) Pitch-Pointing CGT/PI WRT TM(S3)$'$ 171

6-18. AFTI(S3,A2) Pitch-Pointing PI Regulator . . 173

6-19. AFTI(S3,A2) Pitch-Pointing CGT/PI 174

6-20. AFTI(S3,A2) Pitch-Pointing CGT/PI Control Surface Deflections 175

6-21. AFTI(S3,A2,G3) Pitch-Pointing CGT/PI with Respect to Truth Model AFTI(S4,A2,G3) . . . 178

6-22. AFTI(S3) Pitch Rate CGT/PI 181

6-23. AFTI(S3) Pitch Rate CGT/PI Control Surface Deflections 182

6-24. AFTI(S3) Pitch Rate CGT/PI WRT TM(S3)$^+$ . . . 184

6-25. AFTI(S3) Pitch Rate CGT/PI WRT TM(S3)$^-$ . . . 185

6-26. AFTI(S3) Pitch Rate CGT/PI WRT TM(S3)$'$ . . . 186

6-27. AFTI(S3,A2) Pitch Rate CGT/PI 188

6-28. AFTI(S3,A2) Pitch Rate CGT/PI--Alternate PI Regulator 190

6-29. True(1) and Computed(2) Standard Deviations of Estimation Error in θ 194

6-30. True(1) and Computed(2) Standard Deviations of Estimation Error in α 195

6-31. True(1) and Computed(2) Standard Deviations of Estimation Error in q 196

6-32. True(1) and Computed(2) Standard Deviations of Estimation Error in α_g' 197

Figure Page

6-33. True(1) and Computed(2) Standard Deviations of Estimation Error in α_g 198

6-34. True(1) and Computed(2) Standard Deviations of Estimation Error in q_g 199

List of Tables

Table Page

6-1. AFTI/F-16 Data at Flight Condition (Mach=0.8, Altitude=10000. Feet) 140

Abstract

This study develops a computer program for interactive execution to aid in the design of Command Generator Tracker control systems employing Proportional-plus-Integral inner-loop controllers and Kalman Filters for state estimation (CGT/PI/KF controllers). Design parameters are specified in the continuous-time domain and the computer program obtains the corresponding discrete-time parameters and determines a direct digital design for sampled-data implementation. Designs are based upon the Linear system model, Quadratic cost, and Gaussian noise process (LQG) assumptions of optimal control theory.

The report discusses the theoretical background and applications of optimal model-following designs which preceded the CGT theory. A development of the CGT/PI/KF controller theory is presented, and performance evaluation tools for the controller design are discussed. Following a brief description of the computer program developed, results of applying it to example aircraft-related controller design problems are presented and discussed. Among the designs presented are controllers for conventional pitch rate and decoupled pitch-pointing control for an aircraft system model representative of modern aircraft longitudinal dynamics. The CGT/PI/KF controller is found to be a

technique particularly well suited to the typical aircraft control design problem wherein a multi-input multi-output (MIMO) system is to have specified output response behavior to commanded inputs ("handling qualities") while simultaneously rejecting disturbances of specifiable characteristics.

The computer program is fully documented in the appendices of the report. Included are a "Programmer's Manual," a "User's Manual," sample program input and output, a program listing, and a listing of job control language required to obtain an executable object file. These pertain to the computer program as implemented on a Control Data Corporation CYBER computer system and interactive execution under INTERCOM.

DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS
VIA COMMAND GENERATOR TRACKER (CGT)
SYNTHESIS METHODS

I. Introduction

1.1 Background

Modern aircraft designs entail increasingly stringent and complex control requirements. Newer aircraft employ digital flight control systems, utilize multiple control surfaces in each axis, and demand highly refined control characteristics--both in coupled and decoupled command modes.

Early flight control designs initially served pilot relief functions exclusively. Later control systems were designed to improve aircraft stability and control characteristics, but had limited control authority. Such systems were designed in an evolutionary fashion, with considerable trial and error, and generally dealt with multiple control surfaces in an ad hoc fashion.

While modern control design techniques, as exemplified by optimal control theory, showed considerable promise in application to flight control problems during the early and mid-1960s, they have not been adopted by the aircraft designers. The optimal control techniques have been seen as suffering from among the following deficiencies:

traditional design criteria are not readily specified directly in the performance index; it is difficult to select appropriate weighting matrices for the cost functionals to achieve the desired response; the resulting controllers require full-state feedback but measurements of all states are generally not available, so the designs must be reduced or approximated for implementation (or filters or observers must be added); and the typical formulation of the optimal controller solves only the regulator problem and not the required tracking response to a command input.

To achieve greater utility in the application of optimal control techniques to flight control problems, a control synthesis technique known as "model-following" has been used. Essentially, the goal of model-following is to control a given system so that its outputs "follow" those generated by a "model" system which represents the desired dynamic behavior. By formulating the optimization to achieve model-following, the difficulties associated with defining an appropriate performance index and selecting quadratic weighting matrices have been alleviated. But other problems have persisted: the resulting designs still require full-state feedback; it is difficult to achieve desired response behavior to disturbances affecting the system; and the model-following achieved is only for system response to initial conditions with no inputs. Extensions of the theory to allow model-following for forced response to command inputs have been developed but require that the input itself be specified by a dynamic model.

Recent developments in modern control theory have been unified into a new synthesis technique for model-following referred to as Command Generator Tracking. The controller is designed so that the system outputs follow the trajectories prescribed by a command generator, while simultaneously rejecting disturbances with specified characteristics. The model generating commands as a prefilter to the pilot inputs may incorporate the desired closed-loop dynamics and tracking characteristics, and the controller can take full advantage of all available control surfaces appropriate to the control task with systematically determined crossfeed gains as well as single-channel gains.

As discussed in this thesis, the Command Generator Tracking control system is designed as a digital controller composed of three elements: a Command Generator Tracker (CGT) processing command inputs to define inputs to the system; a Proportional-plus-Integral (PI) regulator acting as an inner-loop controller to drive the system to follow the CGT inputs; and a Kalman Filter (KF) providing estimates of the system and disturbance states needed by the controllers. The availability of digital computers of a size, speed, capacity, and reliability appropriate to flight control tasks makes digital controller designs feasible. The CGT/PI/KF controller is a direct digital design from continuous-time specifications, especially suited to the modern digital flight control capability.

The design technique for the CGT/PI/KF controller retains the desirable qualities of the earlier model-following techniques. Furthermore, it provides directly for prescribed response to disturbances, does not require that the command input have specified dynamics, and inherently incorporates the state estimation needed to implement the control in the face of only incomplete and noise-corrupted measurements being available instead of all states.

1.2 Problem

The primary objectives of this thesis are:

1. To develop an interactive, user-oriented computer program to aid in the design of CGT/PI/KF controllers. It is to be applicable to arbitrary systems of varying dimensions. Systems and design parameters are specified for a continuous-time problem representation but the controller is a direct digital design.

2. To apply the design program to an aircraft flight control design problem in order to evaluate characteristics of the CGT/PI/KF design technique and qualities of the resulting controller designs.

1.3 Sequence of Presentation

The results of this effort are fully documented in the body of the thesis and in the appendices. The thesis and its appendices, while integral, are intended to serve as separable entities. The main chapters of the thesis consider the theoretical aspects of the CGT/PI/KF design technique and demonstrate practical application of the technique using the computer program developed in this thesis effort. The appendices specifically document the design computer program to allow understanding of its code and operation, as well as its successful application to design problems different from those considered in the thesis.

The body of the thesis report, composed of Chapters II to IV, discusses the background of model-following control designs, the theoretical development of the CGT/PI/KF controller, and the evaluation tools employed for the design. These chapters are followed by Chapter V which presents a general description of the computer program which was developed, and Chapter VI discussing the CGT/PI/KF controller designs achieved for several design problems through use of the program. A final chapter offers conclusions and recommendations for further research.

The first two appendices are guides to the understanding and use of the program: the first is a detailed description of the computations performed by the program and the computer source code; the second describes the operation of the program from a user's perspective, including discussion of specific items of input/output. These are followed by Appendix C which illustrates sample input and output from an execution of the program, Appendix D which contains a full source listing of the computer code, and a final appendix which shows the job control language needed to obtain an executable program file.

II. Model-Following Control

2.1 Introduction

The control philosophy employed in the CGT/PI/KF controller is not new. Since the early 1960s there has been work on the class of optimal controllers referred to as "model-following systems." Early work by Kalman was unified by Tyler in 1964 (Ref 34) and presented as a design method appropriate to aircraft control problems. From 1964 through about 1977, various articles in the technical journals developed extensions to the methods discussed by Tyler and likewise applied them to aircraft control design. The CGT/PI/KF design method was reported first in 1978 (Ref 8). While it is clearly in the class of model-following controllers and has characteristics closely related to those typical of the earlier model-following designs, the theory from which it is derived is distinctly not in a lineal path with the work of the early 1960s. In one consistent development it incorporates all the capabilities of the various model-following designs of the 1964-1977 time frame, provides new capabilities, and does so in a single unified controller/filter structure. Before presenting the theoretical development of the CGT/PI/KF controller, it is appropriate to outline the theoretical bases of the earlier model-following designs briefly and to discuss some of the various extensions and applications which have occurred.

2.2 Model-Following Control in Aircraft Design

The design objective in employing model-following control is to achieve a control design which forces the output behavior of a given system to be like that of a "model" system. This approach to design of optimal controllers has been previously pursued in aircraft control applications to design controllers yielding either "desirable" characteristics or characteristics like those of another aircraft (e.g., in-flight simulators such as the Calspan TIFS aircraft) (Ref 1).

It has been common to consider aircraft dynamic motion as decoupled between the longitudinal and the lateral axes. In each axis the dominant dynamic modes of motion have been characterized by second-order response models--the "short-period" and "dutch-roll" modes of the longitudinal and lateral axes, respectively. Corresponding to these characterizations, it has been common to develop standards of desirable aircraft control behavior according to specifications on a time-response modeled as second-order (Ref 12). With the advent of so-called "decoupled" modes of motion, it has also become common to define the desired decoupled responses in terms of first-order characteristics. Thus, while the aircraft itself may entail dynamics adequately depicted by a model of relatively high order, design specifications and criteria are framed in terms of low-order models. In parallel with this use of low-order models for control synthesis, it has been common to develop the specifications and "handling criteria" for aircraft control from experiments in which a given aircraft is constrained to exhibit dynamics either like those of another aircraft or like those to be examined for goodness or badness of control quality (Refs 34, 35).

Thus the model-following techniques find natural application in the control problems typical of aircraft. For control synthesis, the designer seeks to achieve a controlled response like that of a first- or second-order system with specific attributes. For evaluation and development of controllers and control specifications, the experimenter seeks behavior of a type postulated as different from that inherent for the test aircraft.

2.3 Types of Model-Following Controllers (Refs 26, 34)

Two different techniques for achieving and implementing model-following controllers were developed in the early 1960s and were presented formally by Tyler (Ref 34) in 1964. One technique is referred to as a "model-in-the-performance-index" controller, or more simply as an "implicit" model-following controller. The other is referred to as a "model-in-the-system" controller, or more simply (and in contrast to the first type) as an "explicit" model-following controller.

For the implicit scheme, the model is employed only in design to determine feedback gains for the system states. For the explicit scheme, the model is used not only to determine necessary feedback gains but also feedforward gains on the model states themselves, making it necessary to include a simulation of the model dynamics in the controller. Although it is derived differently, the CGT/PI/KF controller uses an explicit type of implementation.

As originally developed, both the implicit and explicit model-following controllers used models with state dynamics but no inputs (Refs 23, 34). The responses which were matched for system and model thus were the responses to initial conditions. Later work sought to include response to inputs, but this generally then further required specification of the input for which matching was desired (Refs 1, 20, 26, 27).

2.3.1 Implicit Model-Following Controller

(Refs 23, 26, 34). In implicit model-following, the model-following objective is pursued by employing the feedback gains of the optimal controller to modify the coefficients of the open-loop system matrix so that they approach those of the model. This is achieved by including the model in a performance index of a form suggested by Kalman as an alternative to the usual index that weights system state or output deviations only (Ref 22).

For the system

$$\underline{\dot{x}} = \underline{Ax} + \underline{Bu} \tag{2-1}$$

with output

$$\underline{y} = \underline{Cx} \tag{2-2}$$

A standard performance index is of the form

$$J = \frac{1}{2}\int_{t_0}^{\infty} [\underline{x}^T\underline{Qx} + \underline{u}^T\underline{Ru}]\,dt \tag{2-3}$$

which weights state and input deviations and for which the matrix $\underline{Q}$ may be due to weights directly on the states or derived from weightings on the outputs and given by

$$\underline{Q} = \underline{C}^T\underline{Q}_y\underline{C} \tag{2-4}$$

The infinite terminal time plus time-invariance of the system and cost weighting matrices are used to generate a constant-gain steady-state controller.

Instead of the index of equation (2-3), define a performance index which weights the error between system output derivatives and the model dynamics where the model is

$$\underline{\dot{x}}_m = \underline{A}_m\underline{x}_m \tag{2-5}$$

and the corresponding performance index is

$$J_I = \int_{t_0}^{\infty} [(\underline{\dot{y}}-\underline{A}_m\underline{y})^T\underline{Q}_I(\underline{\dot{y}}-\underline{A}_m\underline{y})+\underline{u}^T\underline{Ru}]\,dt \tag{2-6}$$

and the dimension of the output vector $\underline{y}$ and the model state vector $\underline{x}_m$ are the same and the weighting matrix $\underline{Q}_I$ weights errors between the output and model dynamics. This index can be rewritten using equations (2-1) and (2-2) as

$$J_I = \int_{t_0}^{\infty} [(\underline{CAx}+\underline{CBu}-\underline{A_m Cx})^T \underline{Q}_I (\underline{CAx}+\underline{CBu}-\underline{A_m Cx}) + \underline{u}^T \underline{Ru}]\,dt \tag{2-7}$$

Carrying through the various matrix multiplications and collecting terms leads to a performance index similar to that of equation (2-3) but with a cross-weighting term relating deviations in $\underline{x}$ and $\underline{u}$ (Ref 26):

$$J_I' = \int_{t_0}^{\infty} [\underline{x}^T \hat{\underline{Q}}_I \underline{x} + 2\underline{u}^T \hat{\underline{S}}\underline{x} + \underline{u}^T \hat{\underline{R}}\underline{u}]\,dt \tag{2-8}$$

where

$$\hat{\underline{Q}}_I = (\underline{CA}-\underline{A_m C})^T \underline{Q}_I (\underline{CA}-\underline{A_m C}) \tag{2-9a}$$

$$\hat{\underline{S}} = \underline{B}^T \underline{C}^T \underline{Q}_I (\underline{CA}-\underline{A_m C}) \tag{2-9b}$$

and

$$\hat{\underline{R}} = \underline{R} + \underline{B}^T \underline{C}^T \underline{Q}_I \underline{CB} \tag{2-9c}$$

Such an index is appealing since, for a definition of quadratic weights $\underline{Q}_I$ and $\underline{R}$ with simple and direct meaning to the designer in terms of the desired aircraft response, potentially complex and not particularly obvious cost weightings result to which the usual optimization techniques may be applied.

This directness of specification of weighting matrices appropriate to the design objectives, along with its inherent simplicity in implementation, has made the implicit model-following technique attractive to aircraft control system designers (Refs 20, 26, 27, 34). By modeling either specific inputs or classes of inputs, it has been possible to extend the technique to provide response matching for systems driven by inputs (Refs 20, 26, 27).

2.3.2 Explicit Model-Following Controller (Refs 1, 20, 26, 27). In explicit model-following, the model-following is pursued by employing feedback gains around the system to make it behave as a "tight" tracker, with feed-forward gains on the model's states providing the reference input. The design is essentially of an optimal tracker with the reference being the output of the linear system defined by the model. The controller gains are determined from optimization employing a performance index which weights the differences between the system and model outputs.

For the system defined by equations (2-1) and (2-2) and the model of equation (2-5), an appropriate performance index is (Ref 26)

$$J_E = \int_{t_0}^{\infty} [(\underline{y}-\underline{x}_m)^T \underline{Q}_E (\underline{y}-\underline{x}_m)+\underline{u}^T \underline{Ru}]\,dt \tag{2-10}$$

with the system output vector $\underline{y}$ and the model state vector $\underline{x}_m$ having the same dimension and the weighting matrix $\underline{Q}_E$ weighting the differences between the system outputs and the model states. This is a useful special case of matching model outputs (here the entire model state is considered the output).

To modify the performance index to achieve the form of equation (2-3), define an augmented system

$$\dot{\underline{x}}_a = \underline{A}_a \underline{x}_a + \underline{B}_a \underline{u} \tag{2-11}$$

with

$$\underline{x}_a = \left[\begin{array}{c} \underline{x} \\ \hline \underline{x}_m \end{array}\right] \tag{2-12a}$$

$$\underline{A}_a = \left[\begin{array}{c|c} \underline{A} & \underline{0} \\ \hline \underline{0} & \underline{A}_m \end{array}\right] \tag{2-12b}$$

and

$$\underline{B}_a = \left[\begin{array}{c} \underline{B} \\ \hline \underline{0} \end{array}\right] \tag{2-12c}$$

For this augmented system the corresponding performance index is

$$J_E' = \int_{t_0}^{\infty} [\underline{x}_a^T \hat{\underline{Q}}_E \underline{x}_a + \underline{u}^T \underline{Ru}]\,dt \tag{2-13}$$

with

$$\hat{\underline{Q}}_E = \left[\begin{array}{c|c} \underline{C}^T\underline{Q}_E\underline{C} & -\underline{C}^T\underline{Q}_E \\ \hline -\underline{Q}_E\underline{C} & \underline{Q}_E \end{array}\right] \tag{2-14}$$

The optimal control input for the augmented system then is

$$\underline{u}^* = -\underline{G}^*_c\underline{x}_a \tag{2-15}$$

where $\underline{G}^*_c$ represents the optimal feedback gain matrix. This can be rewritten in terms of the original system and model states:

$$\underline{u}^* = -[\underline{G}^*_{c_1}\underline{x}+\underline{G}^*_{c_2}\underline{x}_m] \tag{2-16}$$

where $\underline{G}^*_{c_1}$ and $\underline{G}^*_{c_2}$ are, respectively, the feedback gains on the system states and the feedforward gains on the model states. Tyler (Ref 34) demonstrates that the gains $\underline{G}^*_{c_1}$ are independent of the model to be followed while the $\underline{G}^*_{c_2}$ gains depend both on the system and the model.

Because the quality of the model-following achieved depends on the tightness of the inner-loop tracker, the explicit model-following design often entails rather high feedback gains. These high feedback gains along with the greater complexity due to incorporation of the model within the control system has made the explicit scheme less popular for controller synthesis than the implicit scheme, although it has found application in the cases in which an experimental aircraft is to behave with specific dynamic characteristics (Refs 34, 35).

2.4 Comparison of Implicit and Explicit Model-Following

For perfect model following, i.e. system output and model output being equal for all time, Erzberger (Ref 17) showed in 1968 that the implicit and explicit model-following controllers are of equivalent capability if the system is perfectly modeled and no disturbances impinge upon it in actual operation. For either controller, the following must be true (Ref 17):

$$[(\underline{CB})(\underline{CB})^{\dagger}-\underline{I}]\ [\underline{A}_m\underline{C}-\underline{CA}] = \underline{0} \qquad (2\text{-}17)$$

where the superscript $\dagger$ denotes the matrix pseudo-inverse, and where $\underline{A}$, $\underline{B}$, $\underline{C}$, and $\underline{A}_m$ are as defined in equations (2-1), (2-2), and (2-5). This relation was derived from algebraic conditions ensuing from the requirement that system and model outputs be exactly equal. For implicit model-following this exact equality leads to a condition on the range spaces of $(\underline{CB})$ and $[\underline{A}_m\underline{C}-\underline{CA}]$. For explicit model-following the requirement of equation (2-17) follows from constraining the time derivatives of the system and model outputs to be equal at the initial time, and the condition that the control input must be bounded. If equation (2-17) is not satisfied, then neither model-follower can achieve perfect tracking with bounded inputs (Ref 17), and increases in the quadratic weights $\underline{Q}_I$ or $\underline{Q}_E$ of equations

(2-6) or (2-10) will not improve the model-following beyond some minimum for the system-model pair.

But in the general case, and particularly when the system is not perfectly modeled and/or disturbances act on it, the implicit and explicit model-followers have different characteristics. Following is a summary of some of those characteristics discussed in the works of the 1964 to 1977 era and referenced previously in this chapter.

By its very nature of weighting deviations on rates, the implicit scheme places primary emphasis on the controlled system's transient behavior. It does not guarantee matching of steady-state behavior even for the nominal parameter values. The feedback gains on the system tend to be lower than for the explicit scheme, but their values and the ultimate success achievable for the design depend on the initial disparity in the open-loop system matrix and the model matrix ($\underline{A}$ and $\underline{A}_m$, respectively). The relatively low gains and the inherent sensitivity of those gains to the specific values of the matrix coefficients for the system make the implicit controller more sensitive to model inaccuracies and parameter variation than the explicit controller. Also, since the model outputs and system outputs are not actually compared in the controller implementation, the controller is blind to any errors between them and can take no corrective action to align them. However, in contrast to the explicit scheme, since the system dynamics are modified to match those of the model, response of the closed-loop system to random zero-mean disturbances (such as clear-air turbulence) can be made to have desirable characteristics without the need for disturbance states in the controller. Finally, the most significant advantage of the implicit model-follower is its simplicity in implementation, and it is primarily this characteristic which motivated earlier strong interest in it as a control synthesis technique.

Due to the weighting of the difference between system outputs and model outputs, the explicit scheme places relatively greater emphasis on the steady-state behavior than on the transient behavior of the model-following. Thus, for nominal parameter values, a perfect system model, and no disturbances acting on the actual system, the explicit controller will achieve model-following in the steady-state. But for gains of magnitude comparable to those of the implicit controller, the quality of the model-following in the transient phase of response will often be worse for the explicit controller. Since the inner-loop feedback must achieve tight tracking performance, the gains required generally are greater than those needed for the implicit scheme. On the other hand, these higher gains along with the fact that the system outputs and model states are actually compared by the controller make the explicit controller less sensitive to parameter variation, modeling errors, and errors in initial conditions. Model-following is achieved only with respect to states of the model which are driven through defined command input channels. Thus, unless disturbances are specifically incorporated into the model, the closed-loop system response to disturbances will be like that of the inner-loop system, which may not be what the designer wishes. Finally, since the command inputs must first be processed through the model dynamics before becoming the reference input to be tracked there may be some time delay in the system's response to inputs.

2.5 Introduction to the CGT/PI/KF Controller

In the previous section it was seen that the explicit model-following controller structure potentially offers important advantages due to its inherently superior performance in real implementations, i.e. controllers for systems not perfectly known and subject to parameter variation. Its primary disadvantages are its greater complexity and its potentially undesirable performance when the system is subject to random disturbances.

With the readily available digital computers for implementation of real-time digital control laws, the complexity of the controller becomes a less significant issue. Accepting that a greater degree of controller complexity can be accommodated in aircraft designs of the future, the CGT/PI/KF controller's characteristics make it attractive for achieving design objectives.

Like the explicit model-follower discussed in this chapter, the CGT/PI/KF controller employs feedforward gains providing a reference input, but to a proportional-plus-integral inner loop regulator which seeks to maintain the difference between the reference and the system outputs at zero. Both the feedforward and the feedback gains are computed independently without resort to an augmented system description and corresponding augmented Riccati equation. The final gain matrices for the closed-loop controller are then obtained from the two independent solutions and characteristics of the system, using simple matrix multiplication.

Unlike the explicit model-follower, the CGT/PI/KF can readily include models of random disturbances affecting the system. The resulting controller then can be made to reject the corresponding real disturbances, in the stochastic zero-mean sense. Thus, disturbance rejection can be concentrated in those frequency bands where predominant disturbances are expected in actual use.

Since the inner-loop of the CGT/PI/KF controller consists of a PI regulator, the actual system controlled response will achieve model-following in steady-state despite errors in system modeling, parameter variation, or unmodeled constant disturbances. Thus the CGT/PI/KF controller is less sensitive to model definition errors than the earlier explicit scheme.

As is described in Chapter III, the determination of the feedforward gains for the CGT controller makes a single assumption for the command input: the command input is assumed to vary slowly in comparison to the system and model dynamics and thus is approximately constant during the controller sampling interval. With this minor restriction on the command input, feedforward gains on the input, the model states, and the disturbance states can be determined. Moreover, the controller will employ same-cycle feedthrough of the command input both directly to the system input and to the model dynamics update. In this way the potential for delay in system response to inputs is greatly alleviated and transient response is enhanced significantly.

Finally, since the controller inherently incorporates a Kalman filter for estimation of modeled system and disturbance states, all variables needed for controller implementation are available. Moreover, account is thus properly taken of the true stochastic nature of the problem, wherein the system and disturbances may be driven by noises modelable as zero-mean Gaussian random processes, and the measurements of the system may be both incomplete and corrupted by noises modelable as zero-mean Gaussian random processes.

III. CGT/PI/KF Theoretical Development

3.1 Overview of the Theory

The design objective in employing the Command Generator Tracker control system is to constrain a given system so that its output response to commanded inputs follows a model trajectory while rejecting modeled disturbances. Both the model trajectory and the disturbances are derived as outputs of linear system models, with the command model driven by the command inputs.

Although the CGT design does not ensure perfect tracking of the model outputs during the transient phase of response, by formulating the design equations based upon such an idealization, the necessary feedforward gains from the command model states, command inputs, and disturbance states to the system inputs can be readily derived.

The CGT design solution can be formulated as an open-loop design, depending only upon the system, command, and disturbance models. However, if the system is marginally stable or unstable, is sluggish or otherwise ill-behaved in response to inputs, then an inner-loop regulator may be employed. In addition, the system model is often not known with certainty or not modeled in full detail in the design, and unmodeled disturbances may also impinge upon the system. Therefore, the preferred implementation consists of an inner-loop controller employing state feedback to act as a regulator with Proportional-plus-Integral control action (to follow nonzero commands with zero steady-state error), and with the feedforward gains of the CGT providing the translation from command inputs to system inputs.

Moreover, if the system and disturbance states are not all available as needed by the controller, or if noise corrupts the available measurements, then a Kalman filter may be employed for system and disturbance state estimation in the overall controller implementation.

Thus, the controller design to be developed in this thesis consists of a Command Generator Tracker (CGT) providing inputs to the system, a regulator with Proportional-plus-Integral control action (PI) operating on those inputs and the system states so as to drive the system outputs along the model output trajectory, and a Kalman filter (KF) providing disturbance state estimates to the PI controller. A general block diagram showing the resulting CGT/PI/KF controller structure is shown in Figure 3-1.

While the CGT/PI/KF controller can be developed as a continuous-time controller, it is developed here in its discrete-time form. With the increasing availability of small, fast, and rugged computers, many controller designs are being implemented as discrete-time algorithms operating on sampled system measurements and providing discrete-valued inputs to the system through sample-and-hold devices. Advantages of such an implementation include

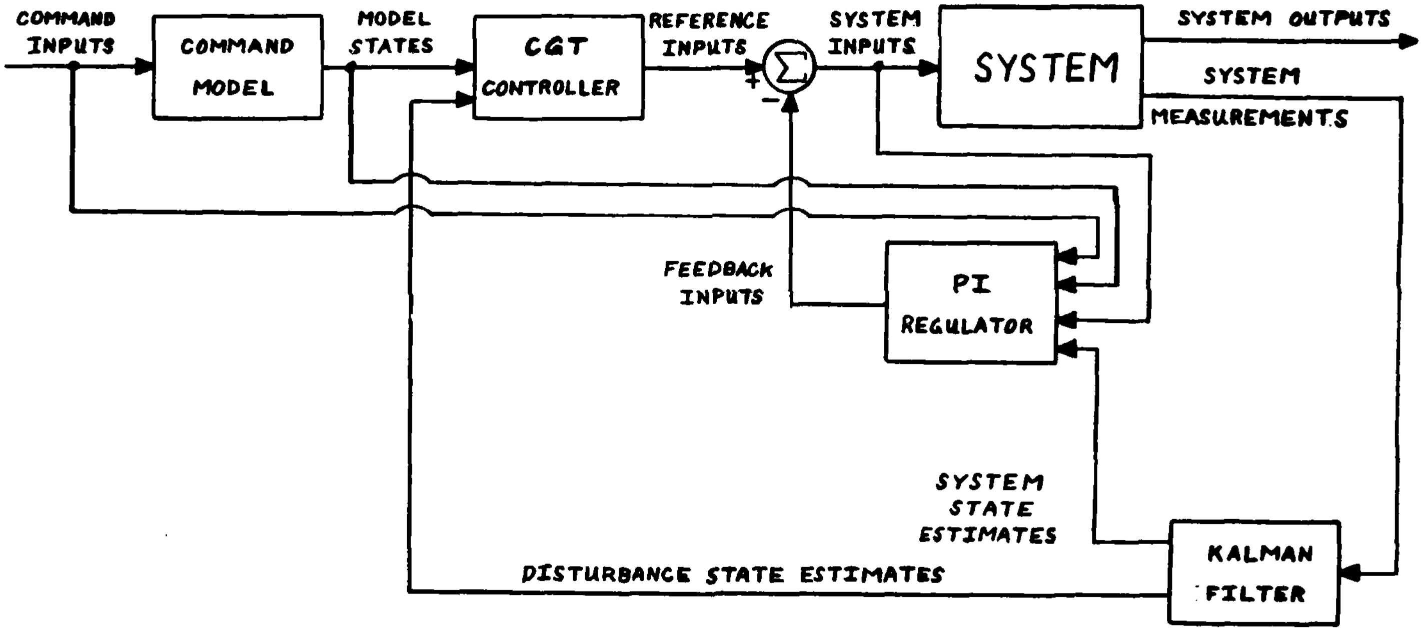

Fig. 3-1. CGT/PI/KF General Block Diagram

greater operating reliability, easier modification of the control laws, and the opportunity to select from among various different controllers each of which may be more sophisticated than feasible for a continuous-time controller.

Although the controller design determines a discrete-time control law, the design itself proceeds from continuous-time specifications. The system to be controlled is generally a continuous-time process and so it is appropriately defined by a continuous-time model. Similarly, the desired system performance is best represented by a continuous-time model. Also, most designers have acquired their design experience from problems which were posed and solved entirely in the continuous-time domain. To take advantage of this experience and in recognition of the fact that the design objectives will be formulated as requirements on the behavior of the system for all time, and not just the sampling instants, the parameters affecting the sampled-data control law determination are derived from the corresponding continuous-time design parameters provided by the designer.

The development here proceeds from the relevant problem description formulated in the continuous-time domain to solution in the discrete-time domain, i.e., a sampled-data controller is synthesized. The elements of this development are derived from the work presented in References 5, 6, 8, 9, and 32.

3.2 System Models

The system for which a CGT/PI/KF controller is to be developed is assumed to be well-represented by a set of linear, time-invariant, stochastic state differential equations with zero-mean white Gaussian noise driving the system and/or disturbance states, and corrupting the system measurements. Such equations generally are derived as linearized perturbation equations for a non-linear system about a nominal operating point. The assumed time-invariance in this context is associated with the derived perturbation model which may be slowly varying but is treated quasi-statically for design. Thus, the system actually is represented by sets of such models, each approximately valid near the operating point at which defined, and control laws are designed for each. The designer may finally formulate the controller gains as functions of various parameters which serve to define the various design point nominal conditions and may implement the control design as a controller of fixed structure but variable gains.

Two system models are employed: a "truth model" and a "design model." The truth model is a model of the system which is as complete and accurate as possible for the control task under consideration. Since the truth model may be of high order, may include states the designer would prefer not to employ for feedback (such as control actuator states), or may include effects the designer considers relatively insignificant, it is often desirable to use a different (generally simpler) model for design. This design model is the basis for the controller and filter gains and provides the set of states to which these gains are applied.

While the quality of the resulting designs is often developed by initially evaluating their performance with respect to the design model, their fine-tuning to final solutions must ultimately be with respect to the system truth model. In addition, even if the designer employs the same model states and parameters in both the design and truth models, the truth model is useful. With the truth model available, the effects of parameter variation may be evaluated since it can be modified while retaining the controller as designed for the nominal parameter values.

3.2.1 Design Model. The design model consists of a system state differential equation, a disturbance state differential equation, an output equation, and a measurement equation.

The system state differential equation is given by

$$\dot{\underset{\sim}{\underline{x}}}(t) = \underline{A}\underset{\sim}{\underline{x}}(t) + \underline{B}\underline{u}(t) + \underline{E}_x\underset{\sim}{\underline{n}}_d(t) + \underline{G}\underset{\sim}{\underline{w}}(t) \qquad (3\text{-}1)$$

where the under-tilde denotes that the variable is modeled as a random process, $\underset{\sim}{\underline{x}}$ and $\underset{\sim}{\underline{n}}_d$ are the Gaussian system and disturbance state vectors respectively, and $\underset{\sim}{\underline{w}}$ is a zero-mean white Gaussian noise with covariance

$$E\{\underset{\sim}{w}(t)\underset{\sim}{w}^T(t+\tau)\} = \underline{Q}\delta(\tau) \qquad (3\text{-}2)$$

where $\delta(\cdot)$ is the delta function and $\underline{u}$ is the system input.

The disturbance state differential equation is

$$\dot{\underset{\sim}{n}}_d(t) = \underline{A}_n\underset{\sim}{n}_d(t) + \underline{G}_n\underset{\sim}{w}_d(t) \qquad (3\text{-}3)$$

where $\underset{\sim}{w}_d$ is a zero-mean white Gaussian noise independent of $\underset{\sim}{w}$ and with covariance

$$E\{\underset{\sim}{w}_d(t)\underset{\sim}{w}_d^T(t+\tau)\} = \underline{Q}_n\delta(\tau) \qquad (3\text{-}4)$$

The output equation is

$$\underset{\sim}{y}(t) = \underline{C}\underset{\sim}{x}(t) + \underline{D}_y\underline{u}(t) + \underline{E}_y\underset{\sim}{n}_d(t) \qquad (3\text{-}5)$$

Finally, the measurement equation is assumed to provide discrete-time measurements given by

$$\underset{\sim}{z}(t_i) = \underline{H}\underset{\sim}{x}(t_i) + \underline{H}_n\underset{\sim}{n}_d(t_i) + \underset{\sim}{v}(t_i) \qquad (3\text{-}6)$$

where the measurement noise $\underset{\sim}{v}$ is zero-mean white Gaussian discrete-time noise independent of both $\underset{\sim}{w}$ and $\underset{\sim}{w}_d$, and of covariance

$$E\{\underset{\sim}{v}(t_i)\underset{\sim}{v}^T(t_j)\} = \underline{R}\delta_{ij} \qquad (3\text{-}7)$$

with δ_{ij} the Kronecker delta function defined as

$$\delta_{ij} = \begin{cases} 0 & \text{when } i \neq j \\ 1 & \text{when } i = j \end{cases} \qquad (3\text{-}8)$$

The dimensionalities for the design model are,

n = number of system states
r = number of system inputs
p = number of system outputs
m = number of system measurements
d = number of disturbance states
w = number of independent system noises
w_D = number of independent disturbance noises

A constraint to be imposed on the dimensionalities is that $p=r$. While a design can be achieved in the more general case by employing the matrix pseudo-inverse, in general it is convenient to consider equal numbers of outputs and inputs.

The dimensions of the various matrices can be inferred from the equations and the above dimensionalities. However, since explicit and clear knowledge of the sizes of all matrices is needed for computer coding of the equations developed in this chapter, appendices dealing specifically with the design computer program fully delineate the sizes of all arrays.

The design model is discretized at a specific fixed controller/filter sampling period T as follows:
The disturbance state description is augmented to the system state model to form

$$\dot{\underline{x}}_a(t) = \underline{A}_a\underline{x}_a(t) + \underline{B}_a\underline{u}(t) + \underline{G}_a\underline{w}_a(t) \qquad (3\text{-}9)$$

where

$$\underset{\sim}{x}_a(t) = \left[\begin{array}{c} \underset{\sim}{x}(t) \\ \hline \underset{\sim}{n}_d(t) \end{array}\right] \quad , \quad \underset{\sim}{w}_a(t) = \left[\begin{array}{c} \underset{\sim}{w}(t) \\ \hline \underset{\sim}{w}_d(t) \end{array}\right] \qquad \begin{array}{r} (3\text{-}9a) \\ (3\text{-}9b) \end{array}$$

and, from equations (3-1) and 3-3),

$$\underline{A}_a = \left[\begin{array}{c|c} \underline{A} & \underline{E}_x \\ \hline \underline{O} & \underline{A}_n \end{array}\right] \tag{3-10a}$$

$$\underline{B}_a = \left[\begin{array}{c} \underline{B} \\ \hline \underline{O} \end{array}\right] \tag{3-10b}$$

$$\underline{G}_a = \left[\begin{array}{c|c} \underline{G} & \underline{O} \\ \hline \underline{O} & \underline{G}_n \end{array}\right] \tag{3-10c}$$

and, from equations (3-2) and (3-4),

$$E\{\underset{\sim}{w}_a(t)\underset{\sim}{w}_a^T(t+\tau)\} = \underline{Q}_a\delta(\tau) \tag{3-10d}$$

where

$$\underline{Q}_a = \left[\begin{array}{c|c} \underline{Q} & \underline{O} \\ \hline \underline{O} & \underline{Q}_n \end{array}\right] \tag{3-10e}$$

The corresponding discrete-time state transition equation is

$$\underset{\sim}{x}_a(t_{i+1}) = \underline{\Phi}_a\underset{\sim}{x}_a(t_i) + \underline{B}_{a_d}\underline{u}(t_i) + \underset{\sim}{w}_{a_d}(t_i) \tag{3-11}$$

where, assuming that $\underline{u}$ is constant over a sample period,

$$\underline{\Phi}_a = e^{\underline{A}_a T} \tag{3-12a}$$

$$\underline{B}_{a_d} = \int_0^T \underline{\Phi}_a(\tau)\ \underline{B}_a d\tau \tag{3-12b}$$

and $\underline{w}_{a_d}$ is zero-mean white Gaussian discrete-time noise of discrete-time noise covariance

$$\underline{Q}_{a_d} = \int_0^T \underline{\Phi}_a(\tau)\underline{G}_a\underline{Q}_a\underline{G}_a^T\underline{\Phi}_a^T(\tau)\,d\tau \tag{3-12c}$$

The matrices of equations (3-12a, b, c) may be partitioned to the original component dimensions to yield

$$\underline{\Phi}_a = \left[\begin{array}{c|c} \underline{\Phi} & \underline{E}_{x_d} \\ \hline \underline{0} & \underline{\Phi}_n \end{array}\right] \tag{3-13a}$$

$$\underline{B}_{a_d} = \left[\begin{array}{c} \underline{B}_d \\ \hline \underline{0} \end{array}\right] \tag{3-13b}$$

$$\underline{Q}_{a_d} = \left[\begin{array}{c|c} \int_0^T [\underline{\Phi}\underline{G}\underline{Q}\underline{G}^T\underline{\Phi}^T + \underline{E}_x\underline{G}_n\underline{Q}_n\underline{G}_n^T\underline{E}_x^T]\,d\tau & \int_0^T \underline{E}_x\underline{G}_n\underline{Q}_n\underline{G}_n^T\underline{\Phi}_n^T\,d\tau \\ \hline \int_0^T \underline{\Phi}_n\underline{G}_n\underline{Q}_n\underline{G}_n^T\underline{E}_x^T\,d\tau & \int_0^T \underline{\Phi}_n\underline{G}_n\underline{Q}_n\underline{G}_n^T\underline{\Phi}_n^T\,d\tau \end{array}\right] \tag{3-13c}$$

Although the matrix of (3-13c) is not generally block diagonal, this does not impact the feedback control that assumes full state knowledge. Invoking the "Certainty Equivalence" property discussed in Section 3.6, the discrete-time models to be used for the deterministic controller design are the system state transition equation defined by

$$\underline{x}(t_{i+1}) = \underline{\Phi}\underline{x}(t_i) + \underline{B}_d\underline{u}(t_i) + \underline{E}_{x_d}\underline{n}_d(t_i) \qquad (3\text{-}14)$$

and the disturbance state transition equation

$$\underline{n}_d(t_{i+1}) = \underline{\Phi}_n\underline{n}_d(t_i) \qquad (3\text{-}15)$$

The corresponding output equation is

$$\underline{y}(t_i) = \underline{C}\underline{x}(t_i) + \underline{D}_y\underline{u}(t_i) + \underline{E}_y\underline{n}_d(t_i) \qquad (3\text{-}16)$$

with matrices $\underline{C}$, $\underline{D}_y$, and $\underline{E}_y$ as in equation (3-5).

Henceforth, all equations relating to the controller design are considered deterministic. As discussed in Section 3.6, the resulting deterministic optimal controller gains are identical to the gains in an LQG stochastic optimal controller in which a Kalman filter provides state estimates from incomplete noise-corrupted measurements rather than assume perfect state knowledge (Refs 2, 32).

3.2.2 Truth Model. For this development, the truth model consists of a state differential equation, a measurement equation, and two equations relating the system and disturbance states of the design model to the truth model states.

Any disturbances which may be considered to impinge upon the system are incorporated into system states for the truth model. The system state model then becomes

$$\dot{\underset{\sim}{\underline{x}}}_t = \underline{A}_t \underset{\sim}{\underline{x}}_t(t) + \underline{B}_t \underline{u}_t(t) + \underline{G}_t \underset{\sim}{\underline{w}}_t(t) \tag{3-17}$$

where $\underset{\sim}{\underline{x}}_t$ is the system state and modeled as a Gaussian random process, $\underset{\sim}{\underline{w}}_t$ is a zero-mean white Gaussian noise with covariance

$$E\{\underset{\sim}{\underline{w}}_t(t) \underset{\sim}{\underline{w}}_t^T(t+\tau)\} = \underline{Q}_t \delta(\tau) \tag{3-18}$$

and $\underline{u}_t$ is the system input.

The measurement equation is assumed to provide discrete-time measurements given by

$$\underset{\sim}{\underline{z}}_t(t_i) = \underline{H}_t \underset{\sim}{\underline{x}}_t(t_i) + \underset{\sim}{\underline{v}}_t(t_i) \tag{3-19}$$

where the measurement noise $\underset{\sim}{\underline{v}}_t$ is zero-mean white Gaussian discrete-time noise independent of $\underset{\sim}{\underline{w}}_t$ with covariance

$$E\{\underset{\sim}{\underline{v}}_t(t_i) \underset{\sim}{\underline{v}}_t^T(t_j)\} = \underline{R}_t \delta_{ij} \tag{3-20}$$

with δ_{ij} as defined by equation (3-8).

The dimensionalities for the truth model are,

n_T = number of system states

r_T = number of system inputs

m_T = number of system measurements

w_T = number of independent noises

Dimensional compatibility for computations necessitates that the number of measurements and inputs be equal for both design and truth models: $m = m_T$ and $r = r_T$.

The additional equations relating the system and disturbance states of the design model to the system states of the truth model are, using a prime to distinguish these from the states of equations (3-1) and (3-3)

$$\underline{x}' = \underline{T}_{DT}\underline{x}_t \tag{3-21a}$$

and

$$\underline{n}_d' = \underline{T}_{NT}\underline{x}_t \tag{3-21b}$$

The truth model is discretized for a specific controller/filter sampling period T, yielding

$$\underline{x}_t(t_{i+1}) = \underline{\Phi}_t\underline{x}_t(t_i) + \underline{B}_{t_d}\underline{u}_t(t_i) + \underline{w}_{t_d}(t_i) \tag{3-22}$$

where, for $\underline{u}$ constant over a sample period,

$$\underline{\Phi}_t = e^{\underline{A}_t T} \tag{2-23a}$$

$$\underline{B}_{t_d} = \int_0^T \underline{\Phi}_t(\tau)\underline{B}_t d\tau \tag{3-23b}$$

and $\underline{w}_{t_d}$ is zero-mean white Gaussian discrete-time noise with covariance

$$\underline{Q}_{t_d} = \int_0^T \underline{\Phi}_t(\tau)\underline{G}_t\underline{Q}_t\underline{G}_t^T\underline{\Phi}_t^T(\tau)\,d\tau \tag{3-23c}$$

3.3 Open-Loop Command Generator Tracker (CGT) (Refs 5, 32)

The command generator to be used here is assumed to be given by a linear, time-invariant state differential equation and an associated output equation which is referred to as the command model. After defining the command model and obtaining its discrete-time equivalent, the concept of the "ideal state trajectory" is introduced and employed to achieve necessary equations for the CGT feedforward gains. These gains assume open-loop implementation. Although such is generally an unsatisfactory implementation, the feedforward gains thus computed are identical to those needed for a system using inner-loop feedback, as is shown subsequently.

3.3.1 Command Model. The command model is defined by a linear, time-invariant state differential equation and an output equation as,

$$\dot{\underline{x}}_m(t) = \underline{A}_m \underline{x}_m(t) + \underline{B}_m \underline{u}_m(t) \qquad (3\text{-}24)$$

and,

$$\underline{y}_m(t) = \underline{C}_m \underline{x}_m(t) = \underline{D}_m \underline{u}_m(t) \qquad (3\text{-}25)$$

The dimensionalities of the command model are,

n_M = number of model states
r_M = number of model inputs
p_M = number of model outputs

Since it is desired to cause the system outputs to follow those of the command model, it is necessary that the number of outputs for each be equal: $p_M = p$.

The discretized command model becomes

$$\underline{x}_m(t_{i+1}) = \underline{\Phi}_m \underline{x}_m(t_i) + \underline{B}_{m_d} \underline{u}_m(t_i) \tag{3-26}$$

$$\underline{y}_m(t_i) = \underline{C}_m \underline{x}_m(t_i) + \underline{D}_m \underline{u}_m(t_i) \tag{3-27}$$

where, for $\underline{u}_m$ constant over a sample period

$$\underline{\Phi}_m = e^{\underline{A}_m T} \tag{3-28a}$$

$$\underline{B}_{m_d} = \int_0^T \underline{\Phi}_m(\tau) \underline{B}_m d\tau \tag{3-28b}$$

and $\underline{C}_m$ and $\underline{D}_m$ are as before in equation (3-25).

3.3.2 Ideal Trajectory. The design objective for the CGT controller is to force the system and command model outputs to be equal,

$$\underline{y}(t_i) = \underline{y}_m(t_i) \tag{3-29}$$

The error in so doing at time t_i is

$$\underline{e}(t_i) = \underline{y}(t_i) - \underline{y}_m(t_i) \tag{3-30a}$$

or, from equations (3-16) and (3-27),

$$\underline{e}(t_i) = [\underline{C}\ \underline{D}_y\ \underline{E}_y] \begin{bmatrix} \underline{x}(t_i) \\ \underline{u}(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} - [\underline{C}_m\ \underline{D}_m] \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \end{bmatrix} \tag{3-30b}$$

To aid in deriving the feedforward gains for the CGT controller, it is useful to formulate an idealization of the solution in which the "ideal state trajectory" and the command model state trajectory are identical for all time. To do so, the deterministic ideal system state and output must be defined so as to satisfy the original system state equation given by equation (3-14)

$$\underline{x}_I(t_{i+1}) = \underline{\Phi x}_I(t_i) + \underline{B}_d\underline{u}_I(t_i) + \underline{E}_{x_d}\underline{n}_d(t_i) \tag{3-31}$$

and the corresponding output equation, equation (3-16):

$$\underline{y}_I(t_i) = \underline{Cx}_I(t_i) + \underline{D}_y\underline{u}_I(t_i) + \underline{E}_y\underline{n}_d(t_i) \tag{3-32}$$

where $\underline{x}_I$ and $\underline{y}_I$ are the ideal state and output vectors, respectively. By definition, the ideal state trajectory must also be such as to maintain zero error between the system and command model outputs:

$$\underline{e}(t_i) = \underline{y}_I(t_i) - \underline{y}_m(t_i) = \underline{0} \tag{3-33}$$

so that

$$\underline{y}_I(t_i) = \underline{y}_m(t_i) \tag{3-34}$$

or, substituting expressions for $\underline{y}_I$ and $\underline{y}_m$ from equations (3-32) and (3-27), respectively, gives

$$[\underline{C}\ \underline{D}_y\ \underline{E}_y] \begin{bmatrix} \underline{x}_I(t_i) \\ \underline{u}_I(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} = [\underline{C}_m\ \underline{D}_m] \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \end{bmatrix} \qquad (3\text{-}35)$$

Since a feedforward control law operating on the command model states and inputs, and the modeled disturbance states, is required for the CGT controller, an additional constraint is imposed for tractability: that the ideal state and input vectors be a linear function of those three vectors. Representing this linear function by partitioned submatrices gives

$$\begin{bmatrix} \underline{x}_I(t_i) \\ \underline{u}_I(t_i) \end{bmatrix} = \begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{13} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} \qquad (3\text{-}36)$$

3.3.3 Open-Loop CGT. The appropriate driving input to the system is the $\underline{u}_I$ vector, so solution for the values of the constant matrices $\underline{A}_{11}$ through $\underline{A}_{23}$ gives the necessary feedforward gain matrices for the open-loop CGT controller. Equations allowing solution for these matrices are developed as follows: using equation (3-31) and augmenting the forward difference expression for the $\underline{x}_I(t_i)$ with the output equation for $\underline{y}_I(t_i)$ of equation (3-32) results in

$$\begin{bmatrix} \underline{x}_I(t_{i+1})-\underline{x}_I(t_i) \\ \underline{y}_I(t_i) \end{bmatrix}$$

$$= \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix} \begin{bmatrix} \underline{x}_I(t_i) \\ \underline{u}_I(t_i) \end{bmatrix} + \begin{bmatrix} \underline{E}_{x_d} \\ \underline{E}_y \end{bmatrix} \underline{n}_d(t_i) \tag{3-37}$$

Now substituting the assumed form for $\underline{x}_I(t_i)$ and $\underline{u}_I(t_i)$ given by equation (3-36) yields

$$\begin{bmatrix} \underline{x}_I(t_{i+1})-\underline{x}_I(t_i) \\ \underline{y}_I(t_i) \end{bmatrix}$$

$$= \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix} \begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{13} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} + \begin{bmatrix} \underline{E}_{x_d} \\ \underline{E}_y \end{bmatrix} \underline{n}_d(t_i) \tag{3-38}$$

The forward difference of $\underline{x}_I(t_i)$ may also be obtained directly from equation (3-36) as

$$[\underline{x}_I(t_{i+1})-\underline{x}_I(t_i)]$$

$$= [\underline{A}_{11} \ \underline{A}_{12} \ \underline{A}_{13}] \begin{bmatrix} \underline{x}_m(t_{i+1})-\underline{x}_m(t_i) \\ \underline{u}_m(t_{i+1})-\underline{u}_m(t_i) \\ \underline{n}_d(t_{i+1})-\underline{n}_d(t_i) \end{bmatrix} \tag{3-39}$$

Assuming $\underline{u}_m$ to be constant (or slowly varying in comparison to the sample period), then

$$\underline{u}_m(t_{i+1}) - \underline{u}_m(t_i) \simeq \underline{0} \tag{3-40}$$

Using the state models for $\underline{x}_m$ and $\underline{n}_d$ of equations (3-26) and (3-15) respectively with driving noises deleted leads to

$$[\underline{x}_I(t_{i+1})-\underline{x}_I(t_i)]$$

$$= [\underline{A}_{11}\ \underline{A}_{12}\ \underline{A}_{13}] \begin{bmatrix} (\underline{\Phi}_m-\underline{I}) & \underline{B}_{m_d} & \underline{0} \\ \underline{0} & \underline{0} & \underline{0} \\ \underline{0} & \underline{0} & (\underline{\Phi}_n-\underline{I}) \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} \tag{3-41}$$

An additional equation for $\underline{y}_I(t_i)$ is obtained from equation (3-27), wherein $\underline{y}_I(t_i) = \underline{y}_m(t_i)$, so

$$\underline{y}_I(t_i) = [\underline{C}_m\ \underline{D}_m] \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \end{bmatrix} \tag{3-42}$$

Performing the interior matrix product of equation (3-41) and augmenting that result with equation (3-42) gives

$$\begin{bmatrix} \underline{x}_I(t_{i+1})-\underline{x}_I(t_i) \\ \underline{y}_I(t_i) \end{bmatrix}$$

$$= \begin{bmatrix} \underline{A}_{11}(\underline{\Phi}_m-\underline{I}) & \underline{A}_{11}\ \underline{B}_{m_d} & \underline{A}_{13}(\underline{\Phi}_n-\underline{I}) \\ \underline{C}_m & \underline{D}_m & \underline{0} \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} \tag{3-43}$$

Equations (3-38) and (3-43) now give two different expressions for the forward difference of $\underline{x}_I(t_i)$ and the output $\underline{y}_I(t_i)$, both involving the desired feedforward matrices. Setting these two expressions equal yields

$$\begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix} \begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{13} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} + \begin{bmatrix} \underline{E}_{x_d} \\ \underline{E}_y \end{bmatrix} \underline{n}_d(t_i)$$

$$= \begin{bmatrix} \underline{A}_{11}(\underline{\Phi}_m-\underline{I}) & \underline{A}_{11}\underline{B}_{m_d} & \underline{A}_{13}(\underline{\Phi}_n-\underline{I}) \\ \underline{C}_m & \underline{D}_m & \underline{0} \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} \tag{3-44}$$

which, after combining terms, gives,

$$\left\{ \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix} \begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{13} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix} - \begin{bmatrix} \underline{A}_{11}(\underline{\Phi}_m-\underline{I}) & \underline{A}_{11}\underline{B}_{m_d} & \underline{A}_{13}(\underline{\Phi}_n-\underline{I})-\underline{E}_{x_d} \\ \underline{C}_m & \underline{D}_m & -\underline{E}_y \end{bmatrix} \right\} \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_i) \\ \underline{n}_d(t_i) \end{bmatrix} = \underline{0} \tag{3-45}$$

Since equation (3-45) must be true for arbitrary $\underline{x}_m$, $\underline{u}_m$, and $\underline{n}_d$ at any sample time, the braced expression must itself be the zero matrix, and thus

$$\begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{13} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix} = \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix}^{-1} \begin{bmatrix} \underline{A}_{11}(\underline{\Phi}_m-\underline{I}) & \underline{A}_{11}\underline{B}_{m_d} & \underline{A}_{13}(\underline{\Phi}_n-\underline{I})-\underline{E}_{x_d} \\ \underline{C}_m & \underline{D}_m & -\underline{E}_y \end{bmatrix} \tag{3-46}$$

Partitioning the indicated matrix inverse into $\underline{\pi}_{ij}$ arrays of the same dimensions as the component arrays gives

$$\underline{\Pi} = \left[\begin{array}{c|c} \underline{\pi}_{11} & \underline{\pi}_{12} \\ \hline \underline{\pi}_{21} & \underline{\pi}_{22} \end{array}\right] = \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix}^{-1} \tag{3-47}$$

where for this development the right-hand side matrix is square since $\underline{y}$ and $\underline{u}$ have the same dimension, and the inverse is assumed to exist. It is possible to generalize this result using matrix pseudo-inverses, but that is not pursued herein (Ref 32). Thus equation (3-46) can be rewritten as

$$\begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{12} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix}$$

$$= \begin{bmatrix} \underline{\pi}_{11} & \underline{\pi}_{12} \\ \underline{\pi}_{21} & \underline{\pi}_{22} \end{bmatrix} \begin{bmatrix} A_{11}(\underline{\Phi}_m-\underline{I}) & \underline{A}_{11}\underline{B}_{m_d} & \underline{A}_{13}(\underline{\Phi}_n-\underline{I})-\underline{E}_{x_d} \\ \underline{C}_m & \underline{D}_m & -\underline{E}_y \end{bmatrix} \tag{3-48}$$

and the explicit partitioned sets of equations are

$$\underline{A}_{11} = \underline{\pi}_{11}\underline{A}_{11}(\underline{\Phi}_m-\underline{I}) + \underline{\pi}_{12}\underline{C}_m \tag{3-49a}$$

$$\underline{A}_{12} = \underline{\pi}_{11}\underline{A}_{11}\underline{B}_{m_d} + \underline{\pi}_{12}\underline{D}_m \tag{3-49b}$$

$$\underline{A}_{13} = \underline{\pi}_{11}\underline{A}_{13}(\underline{\Phi}_n-\underline{I}) - \underline{\pi}_{11}\underline{E}_{x_d} - \underline{\pi}_{12}\underline{E}_y \tag{3-49c}$$

$$\underline{A}_{21} = \underline{\pi}_{21}\underline{A}_{11}(\underline{\Phi}_m-\underline{I}) + \underline{\pi}_{22}\underline{C}_m \tag{3-49d}$$

$$\underline{A}_{22} = \underline{\pi}_{21}\underline{A}_{11}\underline{B}_{m_d} + \underline{\pi}_{22}\underline{D}_m \qquad (3\text{-}49e)$$

$$\underline{A}_{23} = \underline{\pi}_{21}\underline{A}_{13}(\underline{\Phi}_n - \underline{I}) - \underline{\pi}_{21}\underline{E}_{x_d} - \underline{\pi}_{22}\underline{E}_y \qquad (3\text{-}49f)$$

Equations (3-49a) and (3-49c) define solutions for $\underline{A}_{11}$ and $\underline{A}_{13}$ respectively, while the other equations give solutions for the remaining $\underline{A}_{ij}$ matrices which follow from computations involving matrices of known value. These two equations are of the form

$$\underline{X} = \underline{AXB} + \underline{C} \qquad (3\text{-}50)$$

for which an algorithm for solution has been reported (Ref 4) and implemented previously in applications of the CGT design technique (Ref 10). Several conditions must be satisfied in order for a solution to exist. These conditions are (Refs 4, 5)

$$\lambda_{\pi,i} \cdot \lambda_{m,j} \neq 1. \qquad (3\text{-}51a)$$

and

$$\lambda_{\pi,i} \cdot \lambda_{n,j} \neq 1. \qquad (3\text{-}51b)$$

For equations (3-49a) and (3-49c) respectively, where $\lambda_{\pi,i}$ are the eigenvalues of the $\underline{\pi}_{11}$ partition and $\lambda_{m,j}$ and $\lambda_{n,j}$ are the eigenvalues of the matrices $(\underline{\Phi}_m - \underline{I})$ and $(\underline{\Phi}_n - \underline{I})$ respectively. Since the eigenvalues of $\underline{\pi}_{11}$ are related to the inverses of the system transmission zeros, this constraint can be formulated as between the eigenvalues of the command and disturbance models and the system transmission zeros (Refs 5, 14). In addition, no discrete-time transmission zero of the system may equal one.

With $\underline{A}_{11}$ and $\underline{A}_{13}$ determined, and thence all remaining $\underline{A}_{ij}$ partitions, the open-loop Command Generator Tracker control law is obtained from the lower partitioned equation of equation (3-36):

$$\underline{u}_I(t_i) = \underline{A}_{21}\underline{x}_m(t_i) + \underline{A}_{22}\underline{u}_m(t_i) + \underline{A}_{23}\underline{n}_d(t_i) \tag{3-52}$$

The open-loop CGT is implemented as in Figure 3-2.

3.4 Optimal Regulator/PI Controller (Ref 32)

The design goal in employing a Proportional-plus-Integral (PI) controller is to generate a feedback controller which will maintain the system output defined in equation (3-5) at a nonzero commanded value with zero steady state error despite unmodeled constant disturbances which may also drive the system. The idea is well-known in conventional control theory wherein, for a unity feedback configuration, the designer seeks to achieve a forward path transfer function which includes a pole at the origin. This is often achieved by employing a controller with integration of the error in the control variable as a feedback. The resulting design is referred to as a Proportional-plus-Integral controller and the feedback system is described as having the "Type-1 property" (Ref 19) due to the s^1 factor in the forward path characteristic

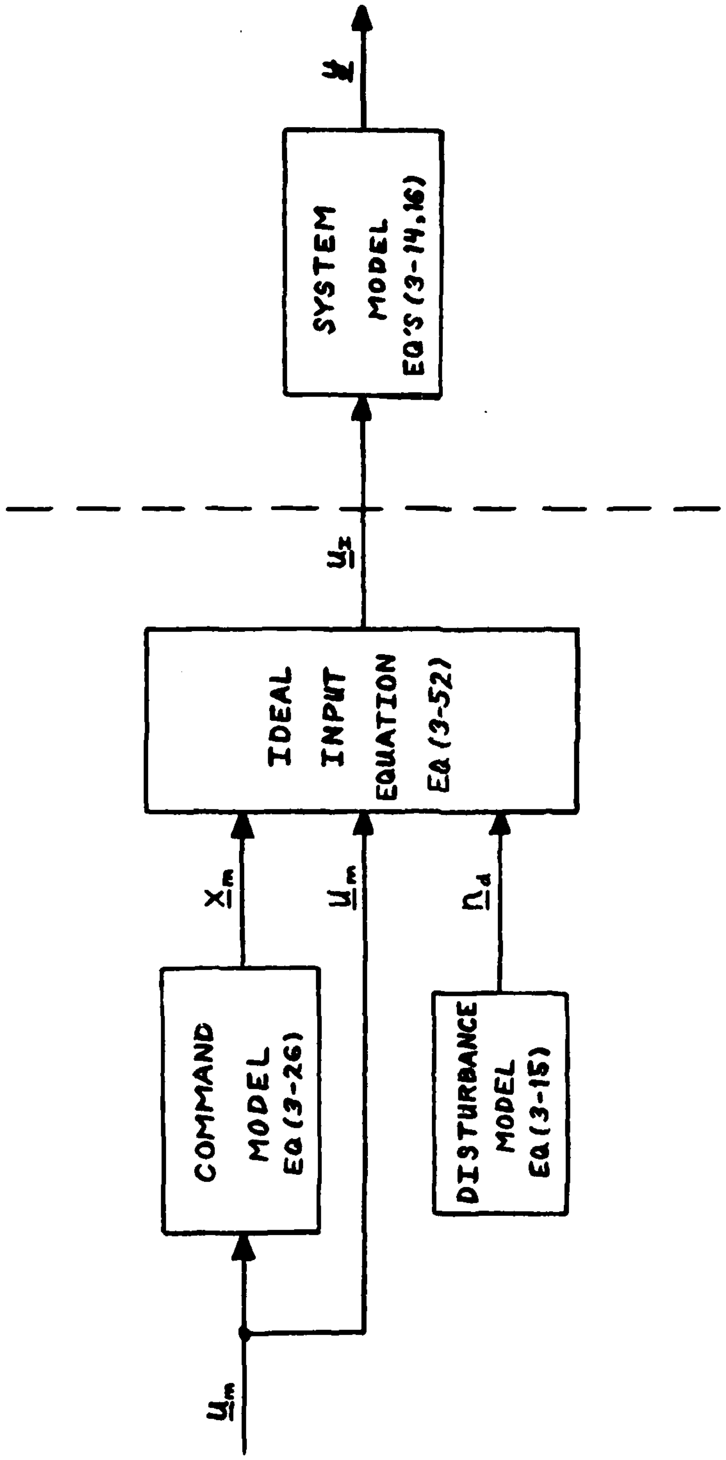

Fig. 3-2. Open-Loop Command Generator Tracker

polynomial (where *s* is the Laplace operator).

In discrete-time state feedback control systems integration is generally approximated based upon the simple Euler integration technique. The PI action can be achieved by performing such pseudo-integration on the regulation error (the differencing of the system and desired outputs) or on the control input rate. The PI controller developed here employs pseudo-integration of the control input rate.

Also, the controller may be formulated for implementation in either of two forms: the "position form" or the "incremental form." The position form represents the current input in its entirety and does so in terms of the total values of the feedback variables. In the incremental form only the change in control input is computed, and it is in terms of changes in the values of the feedback variables since the preceding sample period. The incremental form for the controller has certain advantages over the position form (Ref 7) and is the method for implementation which is developed here.

The optimal PI controller is first developed from a discrete-time problem formulation. Subsequently, the technique for translating a continuous-time quadratic cost formulation to the appropriate discrete-time cost function is demonstrated.

3.4.1 Control Difference PI Controller (Ref 32).

The control difference PI controller achieves its integral quality based upon a control rate pseudo-integration. Define the perturbation control variable as

$$\delta\underline{u}(t_i) = \underline{u}(t_i) - \underline{u}_0 \tag{3-53}$$

where $\underline{u}_0$ is the nominal control input to maintain the system at its equilibrium operating point. Then the forward difference for this perturbation control variable is

$$\delta\underline{u}(t_{i+1}) - \delta\underline{u}(t_i) = (\underline{u}(t_{i+1})-\underline{u}_0)-(\underline{u}(t_i)-\underline{u}_0) \tag{3-54}$$

or,

$$\delta\underline{u}(t_{i+1}) = \delta\underline{u}(t_i) + (\underline{u}(t_{i+1})-\underline{u}(t_i)) \tag{3-55}$$

This can be thought of as an update relation for $\delta\underline{u}(t_{i+1})$ based on Euler integration of control rate, since for Euler integration the change in $\underline{u}$ is

$$\Delta\underline{u}(t_i) = \underline{u}(t_{i+1}) - \underline{u}(t_i) \approx T \cdot \dot{\underline{u}}(t_i) \tag{3-56}$$

where T is the controller sample period and $\dot{\underline{u}}(t_i)$ is the time rate-of-change of the input $\underline{u}$ at the sample time t_i.

This $\Delta\underline{u}$ then is defined to be the control difference or "pseudo-rate." Thus equation (3-55) becomes

$$\delta\underline{u}(t_{i+1}) = \delta\underline{u}(t_i) + \Delta\underline{u}(t_i) \tag{3-57}$$

Defining the perturbation state $\delta\underline{x}(t_i)$ as

$$\delta\underline{x}(t_i) = \underline{x}(t_i) - \underline{x}_0 \qquad (3\text{-}58)$$

where $\underline{x}_0$ is the nominal state vector at the desired operating point, and noting that the perturbation state satisfies the same transition equation as in equation (3-14) with the disturbance state and the noise vector deleted and the perturbation control variable as the input vector, an augmented state description can be formed. The control pseudo-rate is considered the input to an augmented perturbation regulator control problem defined for the augmented perturbation state equation

$$\begin{bmatrix} \delta\underline{x}(t_{i+1}) \\ \delta\underline{u}(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \underline{\Phi} & \underline{B}_d \\ \underline{0} & \underline{I} \end{bmatrix} \begin{bmatrix} \delta\underline{x}(t_i) \\ \delta\underline{u}(t_i) \end{bmatrix} + \begin{bmatrix} \underline{0} \\ \underline{I} \end{bmatrix} \Delta\underline{u}(t_i) \qquad (3\text{-}59)$$

For the optimal regulator the cost criterion to be minimized is

$$J = \sum_{i=-1}^{N} \tfrac{1}{2} \begin{bmatrix} \delta\underline{x}(t_i) \\ \delta\underline{u}(t_i) \\ \Delta\underline{u}(t_i) \end{bmatrix}^T \begin{bmatrix} \underline{X}_{11} & \underline{X}_{12} & \underline{S}_1 \\ \underline{X}_{12}^T & \underline{X}_{22} & \underline{S}_2 \\ \underline{S}_1^T & \underline{S}_2^T & \underline{U} \end{bmatrix} \begin{bmatrix} \delta\underline{x}(t_i) \\ \delta\underline{u}(t_i) \\ \Delta\underline{u}(t_i) \end{bmatrix}$$

$$+ \tfrac{1}{2} \begin{bmatrix} \delta\underline{x}(t_{N+1}) \\ \delta\underline{u}(t_{N+1}) \end{bmatrix}^T \begin{bmatrix} \underline{X}_{f_{11}} & \underline{0} \\ \underline{0} & \underline{0} \end{bmatrix} \begin{bmatrix} \delta\underline{x}(t_{N+1}) \\ \delta\underline{u}(t_{N+1}) \end{bmatrix} \qquad (3\text{-}60)$$

where $\underline{X}_{11}$ weights state deviations from the nominal $\underline{x}_0$, $\underline{X}_{22}$ weights control deviations from the nominal $\underline{u}_0$, and $\underline{U}$ weights control pseudo-rates. The weight $\underline{X}_{f_{11}}$ applies to the state deviation at terminal time, and will not be used further since the regulator to be used here is based on the infinite-time steady-state regulator problem. As is discussed in the next section, the weights $\underline{S}_1$ and $\underline{S}_2$ will arise as the continuous-time cost formulation is converted to a discrete-time cost. Finally, the cross weights $\underline{X}_{12}$ between the state and input vectors will be non-zero if the system output equation includes a non-zero $\underline{D}_y$ or may arise due to the discretization of the continuous-time cost. Note the index for the summation begins at -1: this serves to weight the potentially large control difference which may occur at the initial time due to a change in setpoint (Ref 32) (note $\Delta\underline{u}(t_{-1}) = \underline{u}(t_0) - \underline{u}(t_{-1})$ by equation (3-56).

3.4.2 Continuous-Time Cost (Refs 15, 32). Since the system to be controlled is a continuous-time system, and since its behavior is important at all time and not merely at the controller sample times, the cost function appropriate to the regulator design is

$$J = \tfrac{1}{2}\int_{t_0}^{t_{N+1}} \begin{bmatrix} \underline{\bar{x}}(t) \\ \underline{\bar{u}}(t) \end{bmatrix}^T \begin{bmatrix} \underline{W}_{xx} & \underline{W}_{xu} \\ \underline{W}_{xu}^T & \underline{W}_{uu} \end{bmatrix} \begin{bmatrix} \underline{\bar{x}}(t) \\ \underline{\bar{u}}(t) \end{bmatrix} dt \qquad (3\text{-}61)$$

where,

$$\underline{\bar{x}} = \begin{bmatrix} \delta\underline{x} \\ \delta\underline{u} \end{bmatrix} \qquad (3\text{-}62a)$$

$$\underline{\bar{u}} = \Delta\underline{u} \qquad (3\text{-}62b)$$

$$\underline{W}_{xx} = \begin{bmatrix} \underline{X}_{c_{11}} & \underline{X}_{c_{12}} \\ \underline{X}^T_{c_{12}} & \underline{X}_{c_{22}} \end{bmatrix} \qquad (3\text{-}62c)$$

$$\underline{W}_{uu} = \underline{U}_c \qquad (3\text{-}62d)$$

and, for this development the cross weight $\underline{W}_{xu}$ is assumed to be zero, since its inclusion is rarely necessary to achieve control objectives. Note also the problem is posed as deterministic since its true stochastic nature does not impinge on the optimal regulator design due to certainty equivalence. Terms in the residual cost associated with the stochastic driving noise in the stochastic optimal controller cost formulation are independent of the choice of control function applied and thus do not affect the choice of optimal control (Refs 2, 32).

Furthermore, since the CGT design objective is to drive the system so that its output tracks the model output, it is appropriate that the quadratic weighting matrices specified should apply to the system outputs and inputs. Thus defining the weight on output deviations as $\underline{Y}$ and on input magnitudes as $\underline{U}_y$, the components of the $\underline{W}_{xx}$ matrix are obtained as (Ref 15)

$$\underline{X}_{c_{11}} = \underline{C}^T \underline{Y} \underline{C} \tag{3-63a}$$

$$\underline{X}_{c_{22}} = \underline{U}_y + \underline{D}_y^T \underline{Y} \underline{D}_y \tag{3-63b}$$

$$\underline{X}_{c_{12}} = \underline{C}^T \underline{Y} \underline{D}_y \tag{3-63c}$$

where $\underline{C}$ and $\underline{D}_y$ are as defined in equation (3-5) and $\underline{Y}$ and $\underline{U}_y$ are positive semidefinite and positive definite, respectively. Therefore, $\underline{W}_{xx}$ is positive semidefinite while the $\underline{U}_c$ weighting matrix is required to be positive definite. After forming $\underline{W}_{xx}$ according to equation (3-63) the designer may then modify any elements to achieve, for example, weights on some states not included directly in the output definition of the CGT design objective.

In order to employ this continuous-time cost function of equation (3-61) for solution of the discrete-time optimal regulator it is necessary to obtain the corresponding discrete-time cost function. Begin by conceptually dividing the control interval t_0 to t_{N+1} into (N+1) control intervals of duration equal to the intended controller sample period T. The cost can then be expressed as

$$J = \sum_{i=0}^{N} \left(\int_{t_i}^{t_i+T} \tfrac{1}{2}[\underline{\bar{x}}^T(t)\underline{W}_{xx}\underline{\bar{x}}(t) + \underline{\bar{u}}^T(t)\underline{W}_{uu}\underline{\bar{u}}(t)]\,dt \right) \tag{3-64}$$

where $\underline{\bar{u}}(t)$ is assumed constant over a sampling period and $\underline{\bar{x}}(t)$ satisfies

$$\underline{\bar{x}}(t) = \underline{\Phi}_\delta \underline{\bar{x}}(t_i) + \underline{B}_\delta \underline{\bar{u}}(t_i) \tag{3-65}$$

for all $t \varepsilon [t_i, t_i+T)$ and where, from equation (3-59),

$$\underline{\Phi}_\delta = \left[\begin{array}{c|c} \underline{\Phi} & \underline{B}_d \\ \hline \underline{0} & \underline{I} \end{array}\right] \tag{3-66a}$$

$$\underline{B}_\delta = \left[\begin{array}{c} \underline{0} \\ \hline \underline{I} \end{array}\right] \tag{3-66b}$$

The discrete-time cost function then becomes

$$J = \sum_{i=0}^{N} \tfrac{1}{2} \, [\underline{\bar{x}}^T(t_i)\underline{X}_\delta \underline{\bar{x}}(t_i) + \underline{\bar{u}}^T(t_i)\underline{U}_\delta \underline{\bar{u}}(t_i)$$

$$+ 2\underline{\bar{x}}^T(t_i)\underline{S}_\delta \underline{\bar{u}}(t_i)] \tag{3-67}$$

where,

$$\underline{X}_\delta = \int_{t_i}^{t_i+T} \underline{\Phi}_\delta^T(\tau)\underline{W}_{xx}\underline{\Phi}_\delta(\tau)\,d\tau \tag{3-68a}$$

$$\underline{U}_\delta = \int_{t_i}^{t_i+T} [\underline{B}_\delta^T \underline{W}_{xx} \underline{B}_\delta + \underline{W}_{uu}]\,d\tau \tag{3-68b}$$

$$\underline{S}_\delta = \int_{t_i}^{t_i+T} \underline{\Phi}_\delta^T(\tau)\underline{W}_{xx}\underline{B}_\delta\,d\tau \tag{3-68c}$$

with

$$\underline{\Phi}_\delta(\tau) = \left[\begin{array}{c|c} \underline{\Phi}(\tau) & \underline{B}_d(\tau) \\ \hline \underline{0} & \underline{I} \end{array}\right] \tag{3-68d}$$

$$\underline{B}_d(\tau) = \int_0^\tau \underline{\Phi}(\sigma)\underline{B}d\sigma \tag{3-68e}$$

Equation (3-67) may be expressed more compactly:

$$J = \sum_{i=0}^{N} \tfrac{1}{2} \begin{bmatrix} \underline{\bar{x}}(t_i) \\ \underline{\bar{u}}(t_i) \end{bmatrix}^T \begin{bmatrix} \underline{X}_\delta & \underline{S}_\delta \\ \underline{S}_\delta^T & \underline{U}_\delta \end{bmatrix} \begin{bmatrix} \underline{\bar{x}}(t_i) \\ \underline{\bar{u}}(t_i) \end{bmatrix} \tag{3-69}$$

Note that the cross weighting matrix $\underline{S}$ has been introduced into the cost function by the discretization process. In order to obtain an equivalent discrete-time cost function with no cross weighting (to allow use of standard Riccati equation solvers that assume such a form), define a new system (Ref 29)

$$\underline{\bar{x}}(t_{i+1}) = \underline{\Phi}_\delta' \underline{\bar{x}}(t_i) + \underline{B}_\delta \underline{\bar{u}}'(t_i) \tag{3-70}$$

for which

$$\underline{\Phi}_\delta' = \underline{\Phi}_\delta - \underline{B}_\delta \underline{U}_\delta^{-1} \underline{S}_\delta^T \tag{3-71a}$$

and

$$\underline{\bar{u}}'(t_i) = \underline{\bar{u}}(t_i) + \underline{U}_\delta^{-1} \underline{S}_\delta^T \underline{\bar{x}}(t_i) \tag{3-71b}$$

and for which the corresponding cost function to be minimized is

$$J' = \sum_{i=0}^{N} \tfrac{1}{2} \left[\underline{\bar{x}}^T(t_i)\underline{X}'_\delta\underline{\bar{x}}(t_i) + \underline{\bar{u}}'^T(t_i)\underline{U}_\delta\underline{\bar{u}}'(t_i)\right] \qquad (3\text{-}72)$$

with

$$\underline{X}'_\delta = \underline{X}_\delta - \underline{S}_\delta\underline{U}_\delta^{-1}\underline{S}_\delta^T \qquad (3\text{-}73)$$

If the system of equation (3-70) is either controllable or stabilizable, letting $N \to \infty$ leads to a steady state solution of the discrete Riccati equation represented as $\underline{K}'_R$, where $\underline{K}'_R$ satisfies (Ref 15)

$$\underline{K}'_R = \underline{\Phi}'^T_\delta\underline{K}'_R\underline{\Phi}'_\delta$$

$$+ \underline{X}'_\delta - \underline{B}_\delta\underline{K}'_R\underline{\Phi}'_\delta[\underline{U}_\delta + \underline{B}_\delta^T\underline{K}'_R\underline{B}_\delta]^{-1}\ \underline{B}_\delta\underline{K}'_R\underline{\Phi}'_\delta \qquad (3\text{-}74)$$

and the optimal feedback control is

$$\underline{u}^{*\prime}(t_i) = -\underline{G}_c^{*\prime}\underline{\bar{x}}(t_i) \qquad (3\text{-}75a)$$

where

$$\underline{G}_c^{*\prime} = [\underline{U}_\delta + \underline{B}_\delta^T\underline{K}'_R\underline{B}_\delta]^{-1}\ \underline{B}_\delta^T\underline{K}'_R\underline{\Phi}'_\delta \qquad (3\text{-}75b)$$

The corresponding optimal feedback gain matrix for the original state system ($\underline{\bar{x}}$, $\underline{\bar{u}}$) is (Ref 29)

$$\underline{G}_c^* = \underline{G}_c^{*\prime} + \underline{U}_\delta^{-1}\ \underline{S}_\delta^T \qquad (3\text{-}76)$$

Remembering the definition of $\underline{\bar{x}}$ of equation (3-62a) $\underline{G}_c^*$ can be written in partitioned form as

$$\underline{G}_c^* = [\underline{G}_{c_1}^* \,\vdots\, \underline{G}_{c_2}^*] \qquad (3\text{-}77)$$

and the optimal control input is

$$\Delta\underline{u}^*(t_i) = -\underline{G}^*_{c_1}\delta\underline{x}(t_i) - \underline{G}^*_{c_2}\delta\underline{u}(t_i) \tag{3-78}$$

Combining this expression with the definition of the control difference given in equation (3-56) gives

$$\delta\underline{u}^*(t_{i+1}) = \delta\underline{u}^*(t_i) - [\underline{G}^*_{c_1} \mid \underline{G}^*_{c_2}] \begin{bmatrix} \delta\underline{x}(t_i) \\ \delta\underline{u}(t_i) \end{bmatrix} \tag{3-79}$$

3.4.3 Achieving Integral Control. These results do not yet provide the desired integral characteristic for the controller. Such can be achieved by manipulation to a form emulating that of a continuous-time PI controller:

$$\underline{u}^*(t) = -\underline{K}_x\underline{x}(t) + \underline{K}_z\int_{t_0}^{t}[\underline{y}_d - \underline{y}(\tau)]d\tau \tag{3-80}$$

where $\underline{y}_d$ is the desired output and $\underline{y}(t)$ is the actual system output defined in equation (3-5) but with the disturbance term deleted. On the analogy of Euler integration of the tracking error over each sample period, a discrete-time equivalent would be

$$\underline{u}^*(t_i) = -\underline{K}_x\underline{x}(t_i) + \underline{K}_z\sum_{j=-1}^{i-1}[\underline{y}_d - \underline{y}(t_j)] \tag{3-81}$$

This can be expressed equivalently in incremental form, wherein the input at time t_{i+1} is obtained as an update on the input at time t_i:

$$\underline{u}^*(t_{i+1}) = \underline{u}^*(t_i) - \underline{K}_x[\underline{x}(t_{i+1})-\underline{x}(t_i)]$$

$$+ \underline{K}_z[\underline{y}_d-\underline{y}(t_i)] \qquad (3\text{-}82)$$

Returning to the perturbation state notation and noting that the perturbation in output $\underline{y}$ is

$$\delta\underline{y}(t_i) = \underline{y}(t_i) - \underline{y}_d \qquad (3\text{-}83)$$

and

$$\delta\underline{y}(t_i) = \underline{C}\delta\underline{x}(t_i) + \underline{D}_y\delta\underline{u}(t_i) \qquad (3\text{-}84)$$

then

$$\delta\underline{u}^*(t_{i+1}) = \delta\underline{u}^*(t_i) - \underline{K}_x[\delta\underline{x}(t_{i+1})-\delta\underline{x}(t_i)]$$

$$- \underline{K}_z[\underline{C}\delta\underline{x}(t_i) + \underline{D}_y\delta\underline{u}^*(t_i)] \qquad (3\text{-}85)$$

Employing the expression for the forward difference of $\delta\underline{x}(t_i)$:

$$\delta\underline{x}(t_{i+1}) - \delta\underline{x}(t_i) = [\underline{\Phi}-\underline{I}]\delta\underline{x}(t_i)$$

$$+ \underline{B}_d\delta\underline{u}^*(t_i) \qquad (3\text{-}86)$$

allows equation (3-85) to be rewritten as

$$\delta\underline{u}^*(t_{i+1}) = \delta\underline{u}^*(t_i)$$

$$- [\underline{K}_x \quad \underline{K}_z] \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix} \begin{bmatrix} \delta\underline{x}(t_i) \\ \delta\underline{u}^*(t_i) \end{bmatrix} \qquad (3\text{-}87)$$

Since the expressions for $\delta\underline{u}^*(t_{i+1})$ in equations (3-79) and (3-87) are to be equal and employ constant gains, it follows that

$$[\underline{K}_x \quad \underline{K}_z] \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix} = [\underline{G}^*_{c_1} \quad \underline{G}^*_{c_2}] \tag{3-88}$$

and thus that

$$[\underline{K}_x \quad \underline{K}_z] = \begin{bmatrix} (\underline{\Phi}-\underline{I}) & \underline{B}_d \\ \underline{C} & \underline{D}_y \end{bmatrix}^{-1} [\underline{G}^*_{c_1} \quad \underline{G}^*_{c_2}] \tag{3-89}$$

Remembering the matrix composed of $\underline{\pi}_{ij}$ partitions defined by equation (3-47) and writing the partitioned equations explicitly gives values for the feedback matrices of

$$\underline{K}_x = \underline{G}^*_{c_1}\underline{\pi}_{11} + \underline{G}^*_{c_2}\underline{\pi}_{21} \tag{3-90a}$$

and,

$$\underline{K}_z = \underline{G}^*_{c_1}\underline{\pi}_{12} + \underline{G}^*_{c_2}\underline{\pi}_{22} \tag{3-90b}$$

The final equation for the PI regulator implemented in incremental form and for the sample time t_i is,

$$\delta\underline{u}^*(t_i) = \delta\underline{u}^*(t_{i-1}) - \underline{K}_x[\delta\underline{x}(t_i) - \delta\underline{x}(t_{i-1})] + \underline{K}_z[\underline{y}_d(t_i) - \underline{y}(t_{i-1})] \tag{3-91}$$

where now the desired output $\underline{y}_d$ is allowed to be changing and the time indices for $\underline{y}_d$ and $\underline{y}$ differ since the pseudo-integral of the error must include new error introduced at time t_i by the changed value of desired output (Ref 32). Note that this is directly related to the lower limit on the sum being -1 for this formulation.

3.5 Closed-Loop CGT/PI

The developments of Sections 3.3 and 3.4 may now be combined in a closed-loop CGT/PI controller. It will drive the system so as to achieve matching of the actual system's outputs with the command model outputs in steady state despite possible errors in the models used for design and despite unmodeled constant disturbances which may drive the system in addition to those for which rejection was designed.

3.5.1 Perturbation Regulator. Returning to the concept of the ideal trajectory, define the control difference for $\underline{u}_I(t_i)$ as

$$\Delta\underline{u}_I(t_i) = \underline{u}_I(t_{i+1}) - \underline{u}_I(t_i) \tag{3-92}$$

which, using equation (3-52), can be rewritten as

$$\begin{aligned}\Delta\underline{u}_I(t_i) = {} & \underline{A}_{21}[\underline{x}_m(t_{i+1})-\underline{x}_m(t_i)] \\ & + \underline{A}_{22}[\underline{u}_m(t_{i+1})-\underline{u}_m(t_i)] \\ & + \underline{A}_{23}[\underline{n}_d(t_{i+1})-\underline{n}_d(t_i)]\end{aligned} \tag{3-93}$$

Note that henceforth the assumption of equation (3-40), wherein the command model input $\underline{u}_m$ is assumed constant, is not needed.

Define the set of perturbation variables as

$$\delta\underline{x}(t_i) = \underline{x}(t_i) - \underline{x}_I(t_i) \tag{3-94a}$$

$$\delta\underline{u}(t_i) = \underline{u}(t_i) - \underline{u}_I(t_i) \tag{3-94b}$$

$$\delta\underline{y}(t_i) = \underline{y}(t_i) - \underline{y}_I(t_i) \tag{3-94c}$$

and, remembering the definition of $\Delta\underline{u}(t_i)$ given by equation (3-56),

$$\delta\Delta\underline{u}(t_i) = \Delta\underline{u}(t_i) - \Delta\underline{u}_I(t_i) \tag{3-94d}$$

The augmented system perturbation state equation is the same as in equation (3-59) but with $\delta\Delta\underline{u}$ replacing $\Delta\underline{u}$ in that equation. Similarly the steady-state optimal control solution is in equation (3-78) with the $\Delta\underline{u}$ substitution:

$$\delta\Delta\underline{u}^*(t_i) = -\underline{G}^*_{c_1}\,\delta\underline{x}(t_i) - \underline{G}^*_{c_2}\,\delta\underline{u}(t_i) \tag{3-95}$$

Substituting the expression of equation (3-56) for $\Delta\underline{u}(t_i)$ into equation (3-94d) gives

$$\delta\Delta\underline{u}(t_i) = \underline{u}(t_{i+1}) - \underline{u}(t_i) - \Delta\underline{u}_I(t_i) \tag{3-96}$$

Using equation (3-96) to replace $\delta\Delta\underline{u}(t_i)$ in equation (3-95), shifting the time argument backward one sample, and making substitutions for $\delta\underline{x}(t_{i-1})$ and $\delta\underline{u}(t_{i-1})$ with time-shifted versions of equations (3-94a) and (3-94b) respectively gives an equivalent expression for equation (3-95) as

$$\begin{aligned} \underline{u}^*(t_i) - \underline{u}^*(t_{i-1}) - \Delta\underline{u}_I(t_{i-1}) &= -\underline{G}^*_{c_1}[\underline{x}(t_{i-1})-\underline{x}_I(t_{i-1})] \\ &\quad -\underline{G}^*_{c_2}[\underline{u}^*(t_{i-1})-\underline{u}_I(t_{i-1})] \end{aligned} \tag{3-97a}$$

or

$$\begin{aligned} \underline{u}^*(t_i) &= \underline{u}^*(t_{i-1}) + \Delta\underline{u}_I(t_{i-1}) \\ &\quad - \underline{G}^*_{c_1}[\underline{x}(t_{i-1})-\underline{x}_I(t_{i-1})] \\ &\quad - \underline{G}^*_{c_2}[\underline{u}^*(t_{i-1})-\underline{u}_I(t_{i-1})] \end{aligned} \tag{3-97b}$$

3.5.2 Achieving Integral Control. Making substitutions for $\Delta\underline{u}_I$, $\underline{x}_I$, and $\underline{u}_I$ from equations (3-93) and (3-36) respectively and applying appropriate shifts in time argument yields

$$\begin{aligned} \underline{u}^*(t_i) &= \underline{u}^*(t_{i-1}) + \underline{A}_{21}[\underline{x}_m(t_i)-\underline{x}_m(t_{i-1})] \\ &\quad + \underline{A}_{22}[\underline{u}_m(t_i)-\underline{u}_m(t_{i-1})] \\ &\quad + \underline{A}_{23}[\underline{n}_d(t_i)-\underline{n}_d(t_{i-1})] \\ &\quad - \underline{G}^*_{c_1}[\underline{x}(t_{i-1})-\underline{A}_{11}\underline{x}_m(t_{i-1}) \\ &\quad - \underline{A}_{12}\underline{u}_m(t_{i-1}) - \underline{A}_{13}\underline{n}_d(t_{i-1})] \end{aligned}$$

$$- \underline{G}^*_{c_2}[\underline{u}^*(t_{i-1}) - \underline{A}_{21}\underline{x}_m(t_{i-1})$$

$$- \underline{A}_{22}\underline{u}_m(t_{i-1}) - \underline{A}_{23}\underline{n}_d(t_{i-1})] \quad (3\text{-}98)$$

which is in the desired incremental form, but still lacks the integral property characterizing the PI controller. As in Section 3.4 this property can be achieved using a controller of form

$$\delta\underline{u}^*(t_i) = \delta\underline{u}^*(t_{i-1})$$

$$- [\underline{K}_x \quad \underline{K}_z] \begin{bmatrix} \delta\underline{x}(t_i) - \delta\underline{x}(t_{i-1}) \\ \delta\underline{y}(t_{i-1}) \end{bmatrix} \quad (3\text{-}99)$$

Employing the expressions of equations (3-94) and remembering that $\underline{y}_I = \underline{y}_m$ by definition in equation (3-34), equation (3-99) can also be written as

$$\underline{u}^*(t_i) - \underline{u}_I(t_i) = \underline{u}^*(t_{i-1}) - \underline{u}_I(t_{i-1})$$

$$- \underline{K}_x[(\underline{x}(t_i) - \underline{x}_I(t_i)) - (\underline{x}(t_{i-1}) - \underline{x}_I(t_{i-1}))]$$

$$- \underline{K}_z[\underline{y}(t_{i-1}) - \underline{y}_m(t_{i-1})] \quad (3\text{-}100a)$$

or, after rearranging terms,

$$\underline{u}^*(t_i) = \underline{u}^*(t_{i-1}) + [\underline{u}_I(t_i) - \underline{u}_I(t_{i-1})]$$

$$- \underline{K}_x[\underline{x}(t_i) - \underline{x}(t_{i-1})] + \underline{K}_x[\underline{x}_I(t_i) - \underline{x}_I(t_{i-1})]$$

$$+ \underline{K}_z[\underline{y}_m(t_{i-1}) - \underline{y}(t_{i-1})] \quad (3\text{-}100b)$$

Using equation (3-36) and differencing $\underline{x}_I$ and $\underline{u}_I$ at time arguments t_i and t_{i-1} leads to

$$\begin{bmatrix} \underline{x}_I(t_i)-\underline{x}_I(t_{i-1}) \\ \underline{u}_I(t_i)-\underline{u}_I(t_{i-1}) \end{bmatrix} = \begin{bmatrix} \underline{A}_{11} & \underline{A}_{12} & \underline{A}_{13} \\ \underline{A}_{21} & \underline{A}_{22} & \underline{A}_{23} \end{bmatrix} \begin{bmatrix} \underline{x}_m(t_i)-\underline{x}_m(t_{i-1}) \\ \underline{u}_m(t_i)-\underline{u}_m(t_{i-1}) \\ \underline{n}_d(t_i)-\underline{n}_d(t_{i-1}) \end{bmatrix} \tag{3-101}$$

Equation (3-101) allows substitutions for two terms of equation (3-100b) to be made as follows:

$$\begin{aligned} &[\underline{u}_I(t_i)-\underline{u}_I(t_{i-1})] + \underline{K}_x[\underline{x}_I(t_i)-\underline{x}_I(t_{i-1})] \\ = \ &\underline{A}_{21}[\underline{x}_m(t_i)-\underline{x}_m(t_{i-1})] + \underline{A}_{22}[\underline{u}_m(t_i)-\underline{u}_m(t_{i-1})] \\ + \ &\underline{A}_{23}[\underline{n}_d(t_i)-\underline{n}_d(t_{i-1})] \\ + \ &\underline{K}_x\{\underline{A}_{11}[\underline{x}_m(t_i)-\underline{x}_m(t_{i-1})] + \underline{A}_{12}[\underline{u}_m(t_i)-\underline{u}_m(t_{i-1})] \\ + \ &\underline{A}_{13}[\underline{n}_d(t_i)-\underline{n}_d(t_{i-1})]\} \end{aligned} \tag{3-102a}$$

and the right-hand side can be rearranged to yield

$$\begin{aligned} = \ &[\underline{K}_x\underline{A}_{11}+\underline{A}_{21}] \ [\underline{x}_m(t_i)-\underline{x}_m(t_{i-1})] \\ + \ &[\underline{K}_x\underline{A}_{12}+\underline{A}_{22}] \ [\underline{u}_m(t_i)-\underline{u}_m(t_{i-1})] \\ + \ &[\underline{K}_x\underline{A}_{13}+\underline{A}_{23}] \ [\underline{n}_d(t_i)-\underline{n}_d(t_{i-1})] \end{aligned} \tag{3-102b}$$

To ensure consistency in the equations for the ideal state trajectory in the face of changes in the value of the model input and to improve the initial transient performance, it is necessary to apply direct feedthrough of $\underline{u}_m$ to both $\underline{x}_m$ and $\underline{y}_m$ so that equations (3-26) and (3-27) are actually implemented as (Ref 5)

$$\underline{x}_m(t_{i+1}) = \underline{\Phi}_m \underline{x}_m(t_i) + \underline{B}_{m_d} \underline{u}_m(t_{i+1}) \tag{3-26$'$}$$

and

$$\underline{y}_m(t_i) = \underline{C}_m \underline{x}_m(t_i) + \underline{D}_m \underline{u}_m(t_{i+1}) \tag{3-27$'$}$$

where the time argument of $\underline{u}_m$ has been advanced by one sample period. Thus the model states and outputs cannot be updated between sample times, but must await the new command input sample.

3.5.3 CGT/PI Control Law. Using equations (3-27′) and (3-16) with backward time shifts for both and deletion of the disturbance term for the latter gives replacements for $\underline{y}_m$ and $\underline{y}$ in equation (3-100b). Combining these with the replacements of equation (3-102) transforms equation (3-100b) to the final incremental form CGT/PI control law:

$$\begin{aligned}
\underline{u}^*(t_i) = {} & \underline{u}^*(t_{i-1}) - \underline{K}_x[\underline{x}(t_i) - \underline{x}(t_{i-1})] \\
& + [\underline{K}_x \underline{A}_{11} + \underline{A}_{21}]\ [\underline{x}_m(t_i) - \underline{x}_m(t_{i-1})] \\
& + [\underline{K}_x \underline{A}_{12} + \underline{A}_{22}]\ [\underline{u}_m(t_i) - \underline{u}_m(t_{i-1})] \\
& + [\underline{K}_x \underline{A}_{13} + \underline{A}_{23}]\ [\underline{n}_d(t_i) - \underline{n}_d(t_{i-1})] \\
& + \underline{K}_z \left\{ [\underline{C}_m\ \underline{D}_m] \begin{bmatrix} \underline{x}_m(t_{i-1}) \\ \underline{u}_m(t_i) \end{bmatrix} - [\underline{C}\ \underline{D}_y] \begin{bmatrix} \underline{x}(t_{i-1}) \\ \underline{u}^*(t_{i-1}) \end{bmatrix} \right\}
\end{aligned} \tag{3-103}$$

In implementing the CGT/PI control law in this form, it is necessary to consider the initial conditions to provide proper start-up: if any of the system, model, or disturbance states are non-zero at start-up then the initial control to be applied $\underline{u}^*(t_0)$ must be based upon the total values of the state variables at the initial sampling instant. This can easily be achieved by employing the incremental form of equation (3-103) with the states corresponding to index t_i set to the appropriate initial values and the states at index t_{i-1} and all control input vectors must be set to their previous steady state values. At all future sample times, the model states are updated as in equation (3-26′) followed by an update of the control input according to equation (3-103), assuming that system and disturbance state updates are also available as obtained either directly from measurements or as estimates from a Kalman filter update.

3.6 Kalman Filter (Ref 31)

The CGT/PI control law of equation (3-103) entails knowledge of the entire system and disturbance state vectors employed in the design model definition of equation (3-1). Typically, of course, not only are these not all available from direct measurements taken of the system, but the measurements which are available are corrupted by noise. In order to provide estimates of the states a Kalman filter is employed.

Although from the outset of the theoretical development of this chapter the stochastic nature of the design problem has been recognized, it has not formally impacted any of the designs thus far presented. By formulating the problem definition according to the so-called LQG assumptions ("Linear-Quadratic-Gaussian")--a stochastic optimal control design for a system represented by a Linear system model, employing a Quadratic cost criterion, and with Gaussian driving noises, it has been possible to invoke the "Certainty Equivalence" for the controller/filter design (Ref 2). Certainty equivalence states that the optimal stochastic controller for a system designed according to the LQG assumptions consists of an optimal feedback controller designed independently of the stochastic nature of the system and which is equivalent to the corresponding deterministic optimal LQ controller as already discussed, and a Kalman filter for the system, independent of the controller design, to provide the needed state estimates. The controller and filter may thus be designed independently using techniques and computations appropriate to each, then combined and each design further tuned to achieve best overall response for the ensemble configuration.

The development here begins with the definition of the system model for the Kalman filter design. It then proceeds to the time propagation and measurement update equations for the filter covariance and gain, to determination of the steady state Kalman filter gain matrix, and finally to the corresponding state propagation and update equations.

3.6.1 Design Model. The Kalman filter design for use with the CGT/PI controller of equation (3-103) must provide state estimates for both the system and the disturbance of equation (3-1). Therefore the appropriate system model for design is based upon the augmented design model of equation (3-9). The state transition equation is that of equation (3-14) while the measurement equation becomes

$$\underset{\sim}{z}(t_i) = \underline{H}_a \underset{\sim}{x}_a(t_i) + \underset{\sim}{v}(t_i) \qquad (3\text{-}104)$$

where $\underset{\sim}{z}$ and $\underset{\sim}{v}$ are as defined in equation (3-6), $\underset{\sim}{x}_a$ is as defined in equation (3-9a), and $\underline{H}_a$ is given by

$$\underline{H}_a = [\underline{H} \mid \underline{H}_n] \qquad (3\text{-}105)$$

where $\underline{H}$ and $\underline{H}_n$ are as defined in equation (3-6).

3.6.2 Steady-State Kalman Filter (Ref 31). As for the controller discussed previously, the Kalman filter desired for implementation is to employ constant gains. If the augmented design model defined above is stabilizable and detectable then a steady-state solution to the filter Riccati equation can be achieved which determines a corresponding set of filter gains to be used in a constant gain filter. Further, this constant-gain steady-state Kalman filter is asymptotically stable (Ref 32). A constant-gain steady-state filter to be used in conjunction with a constant-gain steady-state controller is motivated here because the system is quasi-static; the filter and controller have short transients at the beginning and end, respectively, of the control period of interest compared to the long time of essentially steady-state performance; and for aircraft applications minimal complexity of the controller/filter is desirable.

In all equations to follow superscripts of "-" and "+" on the time argument for a matrix indicate the value for that matrix at the given time argument is either before or after a measurement update, respectively.

The filter covariance is propagated forward in time using the equation (Ref 31)

$$\underline{P}_a(t_i^-) = \underline{\Phi}_a \underline{P}_a(t_{i-1}^+)\underline{\Phi}_a^T + \underline{Q}_{a_d} \tag{3-106}$$

where $\underline{\Phi}_a$ and $\underline{Q}_{a_d}$ are as defined in equation (3-12) and $\underline{P}_a$ is the Kalman filter's computed covariance matrix for its state estimates.

The measurement update equation for the Kalman filter gain $\underline{K}_F$ is

$$\underline{K}_F(t_i) = \underline{P}_a(t_i^-)\underline{H}_a^T[\underline{H}_a\underline{P}_a(t_i^-)\underline{H}_a^T+\underline{R}]^{-1} \tag{3-107}$$

where $\underline{R}$ is as defined in equation (3-7).

The measurement update equation for the filter covariance is

$$\underline{P}_a(t_i^+) = \underline{P}_a(t_i^-) - \underline{K}_F(t_i)\underline{H}_a\underline{P}_a(t_i^-) \tag{3-108}$$

In steady state the values of $\underline{P}_a(t_i^-)$ and of $\underline{P}_a(t_{i-1}^-)$ will be equal and so, combining equations (3-106) and backward time shifted versions of equations (3-108) and (3-107) gives the steady state expression for the covariance $\overline{\underline{P}}_a$ as

$$\overline{\underline{P}}_a = \underline{\Phi}_a\{\overline{\underline{P}}_a - \overline{\underline{P}}_a\underline{H}_a^T\ [\underline{H}_a\overline{\underline{P}}_a\underline{H}_a^T+\underline{R}]^{-1}\underline{H}_a\overline{\underline{P}}_a\}\underline{\Phi}_a^T + \underline{Q}_{a_d} \tag{3-109}$$

and the steady-state Kalman filter gain $\overline{\underline{K}}_F$ is

$$\overline{\underline{K}}_F = \overline{\underline{P}}_a\underline{H}_a^T\ [\underline{H}_a\overline{\underline{P}}_a\underline{H}_a^T+\underline{R}]^{-1} \tag{3-110}$$

3.6.3. State Estimates. Employing the steady-state Kalman filter gain of equation (3-110), the augmented state vector propagation and measurement update equations are

$$\hat{\underline{x}}_a(t_i^-) = \underline{\Phi}_a\hat{\underline{x}}_a(t_{i-1}^+) + \underline{B}_{a_d}\underline{u}^*(t_{i-1}) \tag{3-111}$$

and

$$\hat{\underline{x}}_a(t_i^+) = \hat{\underline{x}}_a(t_i^-) + \overline{\underline{K}}_F\ [\underline{z}(t_i)-\underline{H}_a\hat{\underline{x}}_a(t_i^-)] \tag{3-112}$$

where $\underline{B}_{a_d}$ is as defined in equation (3-12) and $\underline{u}^*$ is the CGT/PI control input of equation (3-103). Initialization is achieved by setting

$$\hat{\underline{x}}_a(0) = \begin{bmatrix} \overline{\underline{x}}(0) \\ \overline{\underline{n}}_d(0) \end{bmatrix} \tag{3-113}$$

where $\underline{\bar{x}}(0)$ and $\underline{\bar{n}}_d(0)$ are the expected values of the system and disturbance state vectors at the initial time.

3.7 CGT/PI/KF Control Law

The developments of this chapter provide the necessary design equations for the CGT/PI/KF controller and also the equations for implementation.

The elements of the design process consist of (1) designing a CGT controller providing feedforward gains, (2) designing an optimal PI regulator for inner-loop control, and (3) designing a Kalman filter for state estimation needed for application of the controller's gains. While the mutual separability of these three designs is a crucial aspect of the methodology and serves to make a successful CGT/PI/KF design feasible, it must be realized that the quality of the final design depends on the behavior of the three design elements acting in concert. Thus some design tuning is both appropriate and to be expected based upon the final integrated system-controller-filter response.

The final form of the CGT/PI/KF controller is best represented by specifying the equations needed during a typical control input generation cycle. Assume that the controller is transitioning from its having just generated a control input $\underline{u}^*$ at time t_i through the computations necessary to yield the next control input at time t_{i+1}.

Prior to sample time t_{i+1}:

Propagate the augmented state vector forward in time using equation (3-111):

$$\hat{\underline{x}}_a(t_{i+1}^-) = \underline{\Phi}_a \hat{\underline{x}}_a(t_i^+) + \underline{B}_{a_d} u^*(t_i) \tag{3-114a}$$

At sample time t_{i+1}:

Incorporate the new measurement using equation (3-112):

$$\hat{\underline{x}}_a(t_{i+1}^+) = \hat{\underline{x}}_a(t_{i+1}^-) + \overline{\underline{K}}_F \, [\underline{z}(t_{i+1}) - \underline{H}_a \hat{\underline{x}}_a(t_{i+1}^-)] \tag{3-114b}$$

Propagate the command model state vector forward in time using equation (3-26´):

$$\underline{x}_m(t_{i+1}) = \underline{\Phi}_m \underline{x}_m(t_i) + \underline{B}_{m_d} \underline{u}_m(t_{i+1}) \tag{3-115}$$

Compute the new control input using the partitioned state estimates of equation (3-114) and the command model states of equation (3-115) in equation (3-103):

$$\begin{aligned}
\underline{u}^*(t_{i+1}) &= \underline{u}^*(t_i) - \underline{K}_x [\hat{\underline{x}}(t_{i+1}^+) - \hat{\underline{x}}(t_i^+)] \\
&+ [\underline{K}_x \underline{A}_{11} + \underline{A}_{21}] [\underline{x}_m(t_i) - \underline{x}_m(t_{i-1})] \\
&+ [\underline{K}_x \underline{A}_{12} + \underline{A}_{22}] [\underline{u}_m(t_i) - \underline{u}_m(t_{i-1})] \\
&+ [\underline{K}_x \underline{A}_{13} + \underline{A}_{23}] [\hat{\underline{n}}_d(t_{i+1}^+) - \hat{\underline{n}}_d(t_i^+)] \\
&+ \underline{K}_z \left\{ [\underline{C}_m \; \underline{D}_m] \begin{bmatrix} \underline{x}_m(t_i) \\ \underline{u}_m(t_{i+1}) \end{bmatrix} - [\underline{C} \; \underline{D}_y] \begin{bmatrix} \hat{\underline{x}}(t_i^+) \\ \underline{u}^*(t_i) \end{bmatrix} \right\}
\end{aligned} \tag{3-116}$$

IV. CGT/PI/KF Design Evaluation

4.1 Introduction

It was shown in the course of the theoretical development of Chapter III that the CGT/PI/KF design actually consists of three separate designs: the regulator with PI control action, the open-loop CGT or closed-loop CGT/PI controller, and the Kalman filter. Each of these designs is best evaluated according to criteria specifically related to the task that each is to perform.

For the controller, relevant considerations include the closed-loop system poles, the values of the feedforward and/or feedback gains, and the time response of the controlled system states, outputs, and control inputs in either unforced or forced input conditions. For the filter, relevant considerations include the poles of the filter, the values of the filter gains, and the filter's estimation error behavior. The specific elements of these evaluations are discussed in the following sections.

An overall evaluation of the system performance for the CGT/PI/KF closed-loop controller is necessary to tune the entire design properly and judge its ultimate performance. In particular, the evaluation elements suggested here will demonstrate controller performance alone (CGT/PI), as if the needed system and disturbance feedback and feedforward states are available immediately and with perfect accuracy. The true CGT/PI/KF controller will suffer some degree of degraded performance due to the dynamics of the Kalman filter's state estimation and also a slightly increased delay in control generation due to the needed filter computations (Ref 2). These effects may be evaluated and final integrated tuning of the design achieved by employing a performance evaluation structure of the type discussed in Reference 32. Although this thesis effort did not generate the computer coding to perform such a full controller/filter performance evaluation, future efforts will extend this work to accomplish this objective.

4.2 PI Regulator Evaluation

The discrete-time poles of the closed-loop system incorporating the optimal gain $\underline{G}^*_c$ of equation (3-76) and assuming perfect state knowledge are computed from the matrix

$$\underline{\Phi}_{\delta CL} = [\underline{\Phi}_\delta - \underline{B}_\delta \underline{G}^*_c] \qquad (4\text{-}1)$$

with $\underline{\Phi}_\delta$ and $\underline{B}_\delta$ as defined in equation (3-66). The equivalent continuous-time poles are then computed using the inverse of the relation between the z and s transforms

$$z = e^{sT} \qquad (4\text{-}2)$$

in which z and s are respectively the discrete and continuous-time complex poles and T is the controller sample period. This mapping is for the primary strip in the s-plane only and does not consider any possible aliasing effects (Ref 28).

An important consideration in evaluating the controller design is the magnitudes of the feedback gains $\underline{K}_x$ and $\underline{K}_z$ of equation (3-90). Gains which are relatively very small for specific states may indicate that reduction techniques may be usefully applied to the design model. On the other hand, gains that would tend to cause large control inputs may indicate an unsatisfactory design due to the possibility of control saturation or rate limiting.

Although knowledge of the closed-loop poles and examination of the feedback gain matrices does provide insight into the system behavior to be expected, often the most illuminating information can be determined from the system's time response to initial conditions on the states. The time response can be readily simulated using the discretized deterministic state transition equations of either the design or the truth model and driven by the control input

$$\underline{u}^*(t_i) = \underline{u}^*(t_{i-1}) - \underline{K}_x[\underline{x}(t_i) - \underline{x}(t_{i-1})] - \underline{K}_z[\underline{C}\ \underline{D}_y]\begin{bmatrix}\underline{x}(t_{i-1})\\ \underline{u}^*(t_{i-1})\end{bmatrix} \tag{4-3}$$

obtained from equation (3-103) by deleting terms due to the CGT feedforward control. Note that this controller form is valid only for the PI regulator with zero reference input and in response to non-zero initial conditions on the states. Plots of the time histories of the states, the outputs, and the generated control inputs then may be made. These allow evaluation of the quality of the regulation achieved--speed and damping of the state and output response, and the magnitudes and rates of the control inputs actually required.

4.3 CGT or CGT/PI Evaluation

As for the PI regulator evaluation discussed above, consideration of the magnitudes of the feedforward gains may provide useful insights into the CGT design result. For the open-loop CGT the relevant gains are $\underline{A}_{21}$ acting on the command model states and $\underline{A}_{23}$ acting on the disturbance states in the open-loop CGT control law

$$\begin{aligned}\underline{u}(t_i) = \underline{u}(t_{i-1}) &+ \underline{A}_{21}[\underline{x}_m(t_i)-\underline{x}_m(t_{i-1})] \\ &+ \underline{A}_{22}[\underline{u}_m(t_i)-\underline{u}_m(t_{i-1})] \\ &+ \underline{A}_{23}[\underline{n}_d(t_i)-\underline{n}_d(t_{i-1})] \end{aligned} \tag{4-4}$$

obtained from equation (3-103) by setting the PI controller gains $\underline{K}_x$ and $\underline{K}_z$ to zero. For the CGT/PI controller of equation (3-103) the relevant gains are

$$\underline{K}_{x_m} = \underline{K}_x\underline{A}_{11} + \underline{A}_{21} \tag{4-5a}$$

$$\underline{K}_{x_u} = \underline{K}_x\underline{A}_{12} + \underline{A}_{22} \tag{4-5b}$$

and

$$\underline{K}_{x_n} = \underline{K}_x\underline{A}_{13} + \underline{A}_{23} \tag{4-5c}$$

acting on the command model states and inputs, and disturbance states, respectively.

Evaluation of the CGT or CGT/PI control system time response behavior due to step inputs on the command model inputs is crucial to judging the quality of the controller design. Here, plots of the responses of the system's states, outputs, and inputs along with the outputs of the command model allow the designer to evaluate the merits and/or deficiencies of the design.

The time response of the system may be readily simulated using the deterministic state transition equations for the design or truth models, the command model, and the control law of equation (3-103) with $\underline{K}_x$ and $\underline{K}_z$ set to zero for the open-loop CGT or to their PI design values for the CGT/PI.

4.4 Kalman Filter Evaluation

The state equations of the Kalman filter algorithm may be obtained by rewriting equation (3-112) as (Ref 31)

$$\hat{\underline{x}}(t_i^+) = [\underline{I}-\overline{\underline{K}}_F\underline{H}_a]\,\hat{\underline{x}}(t_i^-) + \overline{\underline{K}}_F\underset{\sim}{\underline{z}}(t_i) \tag{4-6}$$

then substituting the expression $\hat{\underline{x}}(t_i^-)$ of equation (3-111) to yield

$$\hat{\underline{x}}(t_i^+) = [\underline{I}-\overline{\underline{K}}_F\underline{H}_a]\ \underline{\Phi}_a\hat{\underline{x}}(t_{i-1}^+)$$

$$+ [\underline{I}-\overline{\underline{K}}_F\underline{H}_a]\ \underline{B}_{a_d}\underline{u}^*(t_{i-1}) + \overline{\underline{K}}_F\underset{\sim}{z}(t_i) \qquad (4\text{-}7)$$

From equation (4-7) it can be seen that the state transition matrix for the filter estimate propagation is

$$\underline{\Phi}_{KF} = [\underline{I}-\overline{\underline{K}}_F\underline{H}_a]\ \underline{\Phi}_a \qquad (4\text{-}8)$$

The filter's discrete-time poles are computed as the eigenvalues of $\underline{\Phi}_{KF}$ and their continuous-time equivalents computed by the method mentioned in Section 4.2. The magnitudes of the Kalman filter gains may be evaluated from the $\overline{\underline{K}}_F$ matrix of equation (3-110). While the filter poles and the magnitudes of the filter gains may provide some useful insights into the filter performance, often the greatest insight is achieved and the most useful filter tuning tool is provided by a covariance analysis (Ref 31).

In the covariance analysis the covariance of the estimation errors of the Kalman filter when applied to the system truth model is propagated forward in time from the initial conditions on the covariances of the truth model states. In parallel with this estimate-error covariance propagation, the filter covariance is itself propagated forward in time. The true and filter computed estimation error covariances may then be compared since the truth model state estimation error covariances can be transformed to errors for the design states using equation (3-21). The designer may then modify the dynamics noise and measurement noise strengths to achieve the desired filter performance: the duration of the initial estimation transient and the steady state covariance obtained. The development of Reference 31 is summarized here to present the necessary equations for the covariance analysis.

Define the augmented state vector $\underline{x}_c$ by

$$\underline{x}_c(t_i) = \left[\begin{array}{c} \underline{x}_t(t_i) \\ \hdashline \hat{\underline{x}}(t_i) \end{array}\right] \tag{4-9}$$

where $\underline{x}_t$ are the truth model states and $\hat{\underline{x}}$ are the filter state estimates. For the filter implemented without impulsive feedback (Ref 31), the augmented state vector satisfies the time propagation relation

$$\underline{x}_c(t_i^-) = \underline{\Phi}_c \underline{x}_c(t_{i-1}^+) + \underline{w}_{c_d}(t_{i-1}) \tag{4-10}$$

where

$$\underline{\Phi}_c = \left[\begin{array}{c:c} \underline{\Phi}_t & \underline{0} \\ \hdashline \underline{0} & \underline{\Phi}_a \end{array}\right] \tag{4-11a}$$

and $\underline{w}_{c_d}$ is zero-mean white Gaussian discrete-time noise of discrete-time noise covariance

$$E\{\underset{\sim}{w}_{c_d}(t_i)\underset{\sim}{w}^T_{c_d}(t_j)\} = \underline{Q}_{c_d}\delta_{ij} \tag{4-11b}$$

with

$$\underline{Q}_{c_d} = \int_0^T \underline{\Phi}_c(\tau)\ \underline{G}_c\underline{Q}_t\underline{G}_c^T\underline{\Phi}_c^T(\tau)\,d\tau \tag{4-11c}$$

and

$$\underline{G}_c = \left[\begin{array}{c} \underline{G}_t \\ \hdashline \underline{0} \end{array}\right] \tag{4-11d}$$

In these equations, $\underline{\Phi}_t$ and $\underline{\Phi}_a$ are from equations (3-23a) and (3-13a) respectively, while $\underline{Q}_t$ and $\underline{G}_t$ are from equations (3-18) and (3-17) respectively.

The measurement update can be represented by

$$\underset{\sim}{x}_c(t_i^+) = \underline{A}_c\underset{\sim}{x}_c(t_i^-) + \underline{K}_c\underset{\sim}{v}_t(t_i) \tag{4-12}$$

in which

$$\underline{A}_c = \left[\begin{array}{c:c} \underline{I} & \underline{0} \\ \hdashline \underline{\bar{K}}_F\underline{H}_t & \underline{I}-\underline{\bar{K}}_F\underline{H}_a \end{array}\right] \tag{4-13a}$$

$$\underline{K}_c = \left[\begin{array}{c} \underline{0} \\ \hdashline \underline{\bar{K}}_F \end{array}\right] \tag{4-13b}$$

and $\underline{\bar{K}}_F$ is the steady state Kalman filter gain, $\underline{H}_a$ is as defined in equation (3-105), and $\underline{H}_t$ and $\underset{\sim}{v}_t$ are as defined in equation (3-19) for the truth model. Initial conditions are provided by

$$\underset{\sim}{x}_c(t_0^+) = \left[\begin{array}{c} \underset{\sim}{x}_t(t_0) \\ \hline \underline{0} \end{array}\right] \qquad (4\text{-}13c)$$

The covariance of the augmented truth state and estimation error is propagated forward from the initial covariance

$$\underline{P}_c(t_0^+) = \left[\begin{array}{c|c} \underline{P}_t(t_0) & \underline{0} \\ \hline \underline{0} & \underline{0} \end{array}\right] \qquad (4\text{-}14a)$$

by

$$\underline{P}_c(t_i^-) = \underline{\Phi}_c \underline{P}_c(t_{i-1}^+)\ \underline{\Phi}_c^T + \underline{Q}_{c_d} \qquad (4\text{-}14b)$$

and

$$\underline{P}_c(t_i^+) = \underline{A}_c \underline{P}_c(t_i^-)\ \underline{A}_c^T + \underline{K}_c \underline{R}_t \underline{K}_c^T \qquad (4\text{-}14c)$$

for which $\underline{P}_t(t_0)$ is the initial covariance matrix for the truth model state initial conditions and $\underline{R}_t$ is the strength of the discrete-time measurement noise for the truth model (equation 3-20)).

Since it will be desired ultimately to consider the estimation error of the design model's system and disturbance states, it is appropriate to define the filter estimation error as

$$\underline{e}_c(t_i^+) = \underline{C}_c \underset{\sim}{x}_c(t_i^+) \qquad (4\text{-}15)$$

with

$$\underline{C}_c = [-\underline{C}_t \mid \underline{I}] \qquad (4\text{-}16a)$$

and

$$\underline{C}_t = [\underline{T}_{DT} \mid \underline{T}_{NT}] \tag{4-16b}$$

where $\underline{T}_{DT}$ and $\underline{T}_{NT}$ are defined in equation (3-21).

At each sample period, the estimation error covariance is thus

$$\underline{P}_e(t_i) = \underline{C}_c \underline{P}_c(t_i^+) \underline{C}_c^T \tag{4-17}$$

and whose diagonal elements are the variances of the estimation error for each system and disturbance state.

The Kalman filter was designed as a constant-gain filter according to the assumption that its computed covariance attains a constant value in steady-state. The filter's computed covariance is given by equation (3-109) and is denoted $\overline{\underline{P}}_a$.

Finally, define the vectors of standard deviations of the true estimation-error and the filter's computed estimation-error for the design model system and disturbance states as $\underline{s}_c$ and $\underline{s}_F$, respectively. The jth element of the time-varying vector $\underline{s}_c$ is,

$$\underline{s}_{c_j} = \sqrt{\underline{P}_{e_{jj}}(t_i)} \tag{4-18a}$$

while the jth element of the constant vector $\underline{S}_F$ is,

$$\underline{s}_{F_j} = \sqrt{\overline{\underline{P}}_{a_{jj}}} \tag{4-18b}$$

Plots of both s_{c_j} and s_{F_j} for each state with respect to time gives information about the Kalman filter performance ideally suited to the needs of achieving a well-tuned design. They demonstrate both the filter's transient and steady state response for each state estimated. Appropriate changes in the state and measurement noise strengths ($\underline{Q}$, $\underline{Q}_n$, and $\underline{R}$ of equations (3-2), (3-4), and (3-7)) can then be made and the corresponding performance re-evaluated (Refs 2, 31).

V. CGT/PI/KF Design Computer Program

5.1 Introduction

The primary objective of this thesis effort is to create a computer program with which to design CGT/PI/KF controllers. This chapter presents a general description of the program which has been developed--hereafter to be referred to as "CGTPIF." More complete descriptions of CGTPIF are given in various appendices to this thesis.

While the specific test application for the program in the context of this thesis has been related to aircraft control design, CGTPIF is written to be applicable to a wide variety of control design problems. It has the following attributes:

1. CGTPIF executes interactively
2. The program utilizes efficient array allocation
    a. Initial memory allocation easily set
    b. Dynamic array allocation within total memory allocated
3. Various modes of entry are possible for the dynamics models
4. Design paths are automatically followed, with user prompts at necessary decision points
5. Requests for input include informative prompts

6. Copious program output is provided

a. Output most relevant to design decisions are provided directly to the terminal

b. Additional detailed output is provided to a separate output file

7. Information relevant to design iteration is preserved

8. Error checking is performed, and messages given as appropriate

CGTPIF employs computational routines available in a library of matrix computer routines described in Reference 24. Exclusive of the library routines, the program has a length of about 2500 lines of source code. The programming language employed is ANSI Standard FORTRAN IV. Although the resulting source code is highly portable, local memory utilization limits for interactive execution may impose constraints. In use on a Control Data Corporation CYBER machine, the necessary load size was achievable with no impact on the source code. Thus the existing source code is in a pure form for whatever system/implementation motivated modifications may be required to achieve interactive load size limits.

5.2 Program Operating Principles and Organization

CGTPIF has three design paths: (1) design of a PI regulator; (2) design of an open-loop CGT or closed-loop CGT/PI controller, and (3) design of a Kalman filter.

AD-A115 510   AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 1/2
DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS VIA COMMAND G--ETC(U)
DEC 81 R M FLOYD
UNCLASSIFIED AFIT/GE/EE/81-20-VOL-1 NL

2 OF 3
AD
A115 510

Corresponding to the first two design options is a controller evaluation set of routines, and corresponding to the third design option is a set of routines for filter evaluation. The evaluation routines perform the computations discussed in Chapter IV.

A general flowchart of CGTPIF is given in Figure 5-1, showing the main execution paths and design entries. The controller sample period is entered, the design model is established, and then the desired design path can be followed. The elementary design path choice is between controller and filter designs. The CGT or CGT/PI and the PI regulator design paths are then options within the controller design. If the PI controller design is pursued prior to the CGT design path, the CGT design will automatically be of the CGT/PI controller. If the PI is not already determined during the current execution of the program then the designer may elect to design either a CGT or CGT/PI controller. However, the CGT controller design is not pursued if the open-loop design model is unstable. The controller design path is followed automatically by the appropriate controller evaluation path. Similarly, the filter design path leads automatically to the filter evaluation. When the evaluation is complete the designer is given the opportunity to loop on the design path, choose a different design path, or terminate program execution. More detailed flowcharts and functional diagrams are given in the appendices.

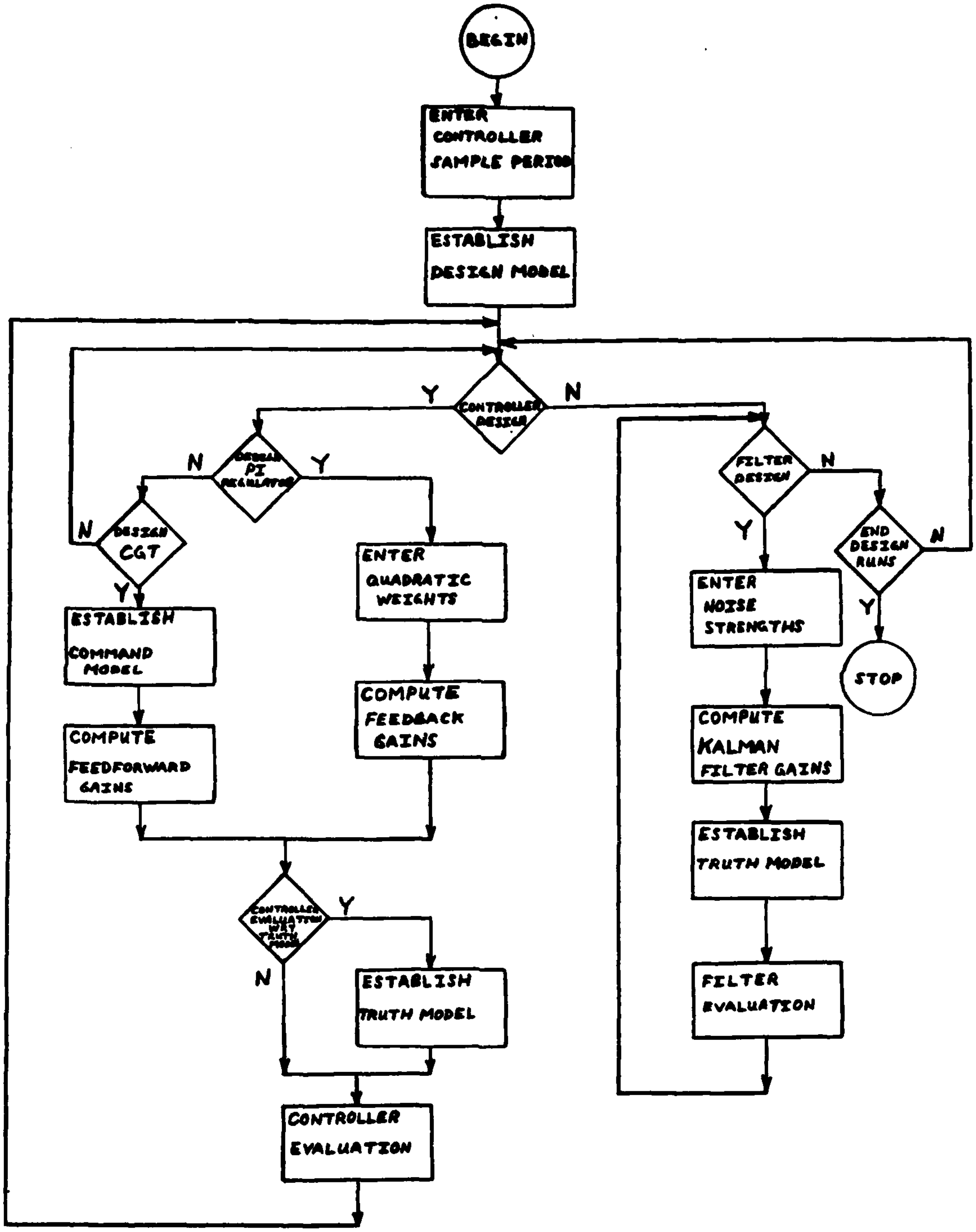

Fig. 5-1. CGTPIF General Flowchart

5.2.1 Interactive Execution. Execution of these options entails both a great deal of program code and memory usage for array storage. The inherent memory requirements are so large that, despite careful coding and efficient array storage techniques, the final program size was much greater than the 65000 octal word limitation of the CYBER interactive system. Thus, in order to achieve interactive operation and provide sufficient free memory for array allocation so that problems of large and variable dimensions could be treated, a CYBER Loader option referred to as "Segmentation" (Ref 13) was employed.

While more detailed information about segmentation is presented in the appendices, a few general comments about it are appropriate here. An attractive feature of segmentation distinguishing it from other methods for achieving selective loading of program elements (such as "Overlays"), is that no modification to the source code is involved in achieving segmentation. A segmented executable object file is produced from a job run which executes job control and segmentation directives operating on the program's compiled object file and any needed object libraries. The job control sequence and segmentation directives are invariant for the kinds of program modifications which may be required to apply CGTPIF to specific design problems. The resulting program executable file then may be run interactively and all program loading and unloading occurs dynamically in the course of execution and is fully transparent to the user.

CGTPIF was written specifically to run interactively. Requests for input, while intentionally brief, tell the user what is expected of him--what, how many, and in what units, as appropriate. Output of information relevant to design decisions to the terminal is compact and automatically provided. Also, the user can determine the amount and category of output to the terminal in some cases, and according to need. These and other characteristics pertaining to CGTPIF's interactive operation are discussed in more detail in succeeding subsections of this chapter.

5.2.2 Array Allocation. It is intended that CGTPIF will not require modification in order to be applied to specific design problems. Named Common blocks are used corresponding to various computational elements of the program. For each Common block an equation is given in Appendix A to allow computation of the minimum total memory allocation needed as a function of the dimensions of the various dynamics models described in Chapter III: the "design," "truth," and "command" models. As currently written, CGTPIF handles problems of orders 15, 20, and 10 for the design, truth, and command models, respectively. In many cases the existing allocations will be satisfactory since any actual usage less than that already allocated for will execute properly. If necessary, allocations can be computed according to the above mentioned equations defining storage requirements. The changed allocations need only be set in the "Main" program routine. All other routines adapt accordingly with no need for modification to the source code.

For each Common block a single array is allocated. The various subroutines of CGTPIF then partition the given allocation as needed to accommodate the specific arrays employed in execution. Each array is thus stored in the minimum memory needed to contain it fully. This method of storage is in contrast to the common technique, wherein each array is individually and explicitly allocated. This dynamic allocation technique instead allows full use of the total allocation for each Common block as needed according to the specific set of dimensionalities employed, and takes no more storage for any individual array than absolutely required to contain it. Thus, for any specific set of dimensions, memory usage is minimized, and a large variety of different dimensionalities can be accommodated within the same total allocation.

5.2.3 Entry of Dynamics Models. Any of the three dynamics models described in Chapter III may be established in any of three ways: (1) the dimensions and matrix elements may be entered directly from the terminal with input prompting from CGTPIF; (2) the dimensions and matrix elements may be entered from a "DATA" file on which the dynamics model from a previous run of the program was written (the writing of such a file entry for each model is a program option); or (3) the dimensions and matrix elements may be established by user provided subroutines.

When entered from the terminal directly or set by subroutines, only non-zero elements of the various matrices need be given. In many cases, this substantially simplifies establishing the matrices and reduces the probability of erroneous entries. Subsequent design runs for the same problem can then simply read the models from the "DATA" file previously created.

5.2.4 Design Paths. Rather than require the user to specify step-by-step the computations to be performed in the design process, the program follows paths automatically and gives prompts at points where options, changes, or inputs are required. Thus, no elaborate or possibly coded list of directives is needed to execute a design.

Each design sequence is followed by entry into the appropriate design evaluation section of code. After evaluation of the specific design, any of the design paths may again be selected, or program execution terminated.

5.2.5 Input Prompts. Each request for input from the user includes a prompt by CGTPIF of information defining the input desired. When an option is offered, the prompt succinctly describes the option and specifies the appropriate form of response (e.g.,"(Y or N)" signifying yes or no, respectively). When a dimension is to be entered, the dimension is identified. When a matrix is to be entered, the matrix is identified, as is its dimensionality, and the appropriate form of input. Similarly, all other requests for input identify the input expected and the form the entry is to take.

However, since writing extensive prompts to the terminal would impede streamlined execution of the program, the messages are brief and require that the user have some understanding of what is involved in achieving each of the designs. The prompts are intended to assist users familiar with the elements of the PI, CGT, CGT/PI, and KF design methods and with the terminology used in this thesis to enter the necessary information for such designs into the program.

5.2.6 Program Output. In the computations involved in the various design paths, a great deal of information is generated. While all the information generated may be relevant to the design, in the usual case only a small fraction of the information is needed to pursue iteration of the design paths.

Information most relevant to achieving the various designs is output to the terminal either automatically or at the user's option. The same information, along with all other potentially useful information generated by the program, is also output to a "LIST" file. After program execution is complete, the user may then look through this file at the terminal (as with "PAGE" command, for example) or may "ROUTE" it to a line printer for a complete listing. All output is given an identifying name or description which is consistent with the terminology of this thesis.

Plots are available as options of the design evaluation routines. The plots produced are of the "line-printer" type. For the controller, plots of selected variables may be output to the terminal, in addition to the full set of plots output to the "LIST" file.

5.2.7 Preservation of Design Information. As discussed previously, the dynamics models, once defined, may be written to a file from which they may be extracted during subsequent runs of the program. In addition, the feedback gains of the PI regulator may be written to the same file for later reuse.

During program execution, user entries in the design iteration are preserved. Thus, for each iteration only those entries to be changed need be given as input, making design iteration both fast and easy in terms of the simple mechanics of the process. Also, computations which are not modified by design iteration are performed only once and the results are preserved for reuse within the current run, as needed.

5.2.8 Error Checking. A common problem encountered in executing computer programs with variable dimension array storage is the unintentional (and often unknown) over-running of the allocated storage area. When this happens, the program may fail (due to the over-writing of program code, for example) or, even worse, the program may appear to run properly but provide erroneous results. To avoid these difficulties, before using each of the Common arrays, CGTPIF computes the allocation needed for the arrays it will generate and compares it with the number of words of memory actually allocated. If more memory is needed than has been provided, a message is written indicating the problem, the Common in question, and the minimum allocation needed. Execution is then aborted. As described in Appendix A, allocations may be changed in the Main routine and a new program created to achieve the necessary array storage.

Error checks for array entry from the terminal are also performed. Not only are the array dimensions identified in prompts requesting entry of arrays, but each entry is checked to verify that it is within the row, column bounds for the array. If not within bounds, the entry is not accepted and a message identifying the problem is given. Also, for matrices requiring special properties (e.g., positive semi-definiteness), entries which clearly violate these requirements (e.g., negative diagonal elements) are not accepted and a message is written to the terminal.

Various other error checks are performed for each input entered from the terminal. These checks ensure that no invalid entries are accepted (e.g., non-positive controller sample time), and no storage out of array bounds occurs. Obviously it is not possible to guard against valid yet erroneous entries, but in most cases the program provides opportunities to correct mistaken entries made and discovered by the user.

Additional tests are performed to ensure dimensional consistency of the dynamic models. These conditions are discussed in Chapter III as each model description is defined. For example, CGTPIF checks that the numbers of inputs and outputs defined for the design model are equal, as well as checking each of the other dimensional conditions as each model is established. If the condition is not met, a message identifying the problem is written and the program execution is aborted.

5.3 Program Usage

CGTPIF was written with the intention of providing a CGT/PI/KF controller design program that is efficient in memory and execution time, streamlined for the user in input and output requirements in interactive operation, and applicable to a variety of problems. These objectives have been achieved.

Preparation for use of the program consists primarily in determining the dimensions of the models to be employed and the specific coefficients of the matrices comprising those models. It is appropriate for the PI design that an initial set of quadratic weights be established also in order to begin the design iteration process.

Assuming that the user understands the nature and requirements of the various design elements, and is familiar with terminology related to such designs, the actual execution of the program involves straightforward response to input prompts. The specific meaning of all program prompts employed is delineated in Appendix B of this thesis. Information most useful to the design iterations is available directly as output to the terminal, while additional information is provided in a separate output file. The terminology employed in the output is defined in Appendix B and related to specific program computations in Appendix A of this thesis.

Results from the use of CGTPIF are presented in the next chapter. More detailed information in the appendices (A through E) include: a "Programmer's Guide," a "User's Guide," sample input and output, a program listing, and a listing of the job card sequence needed to obtain a segmented executable object file of CGTPIF.

VI. CGT Design Results

6.1 Introduction

The program developed in this study has been applied to a variety of control design problems in order to verify its proper functioning and also to evaluate the design process and the characteristics of the controller designs achievable. This chapter discusses several CGT/PI and CGT/PI/KF designs. The purpose is to demonstrate the systematic, logical design process and to show the capabilities achieved by the designs. Although the design iterations are not discussed in detail, a summary of the iterative process is given in Section 6.1.2 below.

6.1.1 Design Examples. The designs discussed here are for two different systems. A simple design example uses a lightly-damped, second-order, single-input-single-output (SISO) system model, with two constant disturbances also driving the dynamics. The second system is an aircraft longitudinal dynamics model with unstable dynamics and with two control inputs. For both systems, CGT designs based upon two different command model dynamics descriptions are developed.

For the simple second-order system, open-loop CGT and closed-loop CGT/PI designs are developed. Command model dynamics representing a first-order system and a well-damped second-order system are used. Specifically, the following designs are discussed: (1) design of open-loop CGT and closed-loop CGT/PI controllers using the first-order command model without disturbances to demonstrate the improvement in model-following achievable with the closed-loop design; (2) two different representations of the second-order command model are used to demonstrate the invariance of controller response with alternative representations of command model dynamics; (3) open-loop CGT and closed-loop CGT/PI designs are developed for the second-order command model dynamics; (4) these are evaluated for the cases of modeled constant disturbances of zero or non-zero unknown magnitude acting on the system dynamics to demonstrate the capabilities of the open-loop implementation and the improvements possible with the closed-loop implementation; and (5) these same controllers are then applied to a second-order system of different dynamics than that used for design to show the affects of modeling error on the performance of the CGT controllers.

For the aircraft longitudinal dynamics system, three different design models are used. All three design models employ simple three-state dynamics models, while one includes actuator states also, and the third includes both actuator and clear-air turbulence states. These are derived as approximations of a truth model which includes a four-state dynamics model and actuator and clear-air turbulence states. Several of these simplified models then serve as design and truth models for specific design cases. The following CGT/PI designs are discussed: (1) design of decoupled pitch controllers for all three design models demonstrate the performance of the CGT/PI controller in achieving decoupled response characteristics and the effects of inclusion or non-inclusion of actuator dynamics in the design model; (2) the decoupling CGT/PI controller for the simplest design model is evaluated with respect to truth models with different parameter values to demonstrate controller performance when subject to modeling errors or parameter variation; (3) design of conventional pitch controllers for the two simplest design models to demonstrate the ability of the CGT/PI design in the case in which fewer system outputs are to be of constrained dynamics than there are independent controls available; and (4) as in item (2) immediately above, the conventional pitch controllers for the simplest design model are evaluated when subject to model errors and parameter variation. Finally, a single Kalman filter design and covariance analysis is discussed. The filter design is based upon the design model which includes turbulence states and is evaluated with respect to the system truth model.

All controller and filter designs are based upon a sample time of 0.02 seconds. Such a sample period is representative of the controller sampling times currently employed in digital flight control systems. Since a direct-digital design is effected, the controller sample period may be set at any value appropriate to the design problem under consideration. This is in contrast to digital designs based upon discrete approximations of continuous controller designs for which the sample period is constrained by the validity of the approximations employed.

In order to facilitate reference to gains employed by the feedforward CGT controller, the following definitions of matrices developed in Chapter III are given: feedforward gain matrices applying to the command model states, command model inputs, and design model disturbance states are referred to as $\underline{K}_{x_m}$, $\underline{K}_{x_u}$, and $\underline{K}_{x_n}$ respectively. Thus for the open-loop CGT these matrices are, from equation (3-52)

$$\underline{K}_{x_m} = \underline{A}_{21} \tag{6-1a}$$

$$\underline{K}_{x_u} = \underline{A}_{22} \tag{6-1b}$$

and

$$\underline{K}_{x_n} = \underline{A}_{23} \tag{6-1c}$$

For the closed-loop CGT/PI the feedforward gain matrices are, from equation (3-116)

$$\underline{K}_{x_m} = \underline{K}_x\underline{A}_{11} + \underline{A}_{21} \tag{6-2a}$$

$$\underline{K}_{x_u} = \underline{K}_x\underline{A}_{12} + \underline{A}_{22} \tag{6-2b}$$

and

$$\underline{K}_{x_n} = \underline{K}_x\underline{A}_{13} + \underline{A}_{23} \tag{6-2c}$$

Similarly, gains of the PI regulator are referred to repeatedly in this chapter. The feedback gain matrices are as given by equations (3-90a) and (3-90b) for $\underline{K}_x$ gains on system states and $\underline{K}_z$ gains on system output errors. The gain matrix for the simple regulator alone is $\underline{G}_c^*$ as given in equation (3-76).

6.1.2 PI Regulator Design for CGT/PI Application.

The design of PI regulators in the context of the CGT/PI controller proves to be straightforward, and is not discussed in detail for the specific design examples. The PI regulator is used to null errors between the reference feedforward inputs provided by the CGT and the system's true state and output response. The PI design objective for the inner-loop implementation is primarily motivated by the need to achieve rapid, well-damped zeroing of these errors. The actual control magnitudes and rates required to achieve such response need not be constrained in the PI design itself to design limit values. The CGT/PI controller generates input magnitudes and rates that often are more a function of the CGT feedforward solutions than of the PI feedback commands alone. Typically, in the closed-loop configuration the most severe input magnitude and rate commands are due to the feedforward gains on the command model inputs. As can be seen in equation (6-2b), the feedforward gain $\underline{K}_{x_u}$ on the command model inputs depends on the PI feedback gain $\underline{K}_x$ (defined in equation 3-90a)) and also on the partitions $\underline{A}_{12}$ and $\underline{A}_{22}$ of the CGT equations (given by equations (3-49b) and (3-49e)).

The procedure which was found useful in the PI-CGT/PI design iteration is described by the following steps:

1. Determine initial estimates of appropriate quadratic weights (for the cost formulation of equation (3-61)) by selecting a set of output errors and corresponding input magnitude and rate maximum admissible values to null the output errors. Output weights are then chosen as the reciprocal of the square of each output error. Similarly, the input magnitude and rate weightings are chosen as the reciprocals of the squares of the maximum magnitude and rate input values. These weightings may be scaled to be symmetrical about unity, with smallest value unity, or however desired for convenience since only the relative values of the weights affect the design results.

2. Using the quadratic weights determined in step 1, a PI regulator is designed and the system response to state initial conditions is evaluated.

3. Often the PI gains are significantly greater than the gains of the simple regulator for the same problem, and the speed and damping of the response is inferior for the given input maximum admissible values. But since the inputs generated in command following with the CGT/PI controller depend in large measure on the feedforward gains, initial iteration of the PI design is directed at achieving good speed and damping of the regulator response despite

possibly exceeding the maximum admissible input values. Thus the weights on output errors are increased and the output behavior in response to initial conditions is evaluated iteratively. At the same time it is often useful to include additional weights on the first derivatives of the outputs to improve the damping. Approximate output derivative weighting can be achieved easily in many cases by introducing weights on states included in the output rate equations. Although the input magnitudes and rates are not specifically constrained by design limits at this stage of the design, it is necessary that judicious adjustment of the relevant quadratic weights be pursued with the objective of achieving the desired output response with minimum control power. Thus it is not appropriate to increase output weightings by orders of magnitude arbitrarily. The designer must instead selectively apply quadratic weights on states and outputs that optimize output regulation with respect to control power.

4. When good regulation has been achieved by iteration on the PI design alone, compute the CGT/PI controller and evaluate its response to command inputs. Difficulties with input magnitudes and rates may then be apparent, and will be most severe during the first few controller sample periods after application of the command due to direct feedforward of the command input.

5. If necessary, iterative design of the CGT/PI controller then proceeds by computing new PI gains to give

improved output regulation and/or reduced control inputs, followed by recomputation of the CGT/PI gains and reevaluation of the system response to command inputs. The goal is to achieve a final design which provides good model-following without exceeding input constraints. This may not be achievable in all cases: sometimes the CGT feedforward gain on the command input is large in itself, and the contribution of the PI gain to it cannot achieve acceptable final feedforward gain (see equation (6-2b)).

6.1.3 Determining Command Models. For single-input-single-output (SISO) system command models for specified dynamics are readily determined (e.g., from a transfer function defining the desired response, a state model may be obtained directly in the "standard controllable form" (Ref 31)). Moreover, any state description yielding the specified command model dynamics will yield identical response to that of any other state description with the same input/output characteristics, for either the open-loop CGT or the closed-loop CGT/PI controller. This is demonstrated in cases to follow.

For multi-input-multi-output (MIMO) systems, determining appropriate command models can be more difficult. In some cases a command model can be obtained from a generalization of the standard controllable form which can be written for a matrix of transfer functions (Ref 19). If the desired output response is defined as decoupled and first-order between each output and input, then command models are easily determined and need include only diagonal matrices. Determination of such first-order, decoupled command models is shown for the aircraft controller design in Section 6.4.5. As in the SISO case, any model which achieves the desired dynamics may be used in the controller with equivalent results.

The design examples discussed here demonstrate the determination of command models. For the flight control example (MIMO) only first-order decoupled response command models are used.

6.1.4 Need for Complete Evaluation Software.

The designs discussed in this chapter are evaluated as open-loop CGT or closed-loop CGT/PI controllers or as independent Kalman filters. Thus, the results demonstrated for the CGT/PI controller are valid strictly for the case of perfect and complete state knowledge. Some degradation in performance, especially with regard to robustness (Ref 32), may accrue due to inclusion of the state estimation provided by a Kalman filter. As discussed in Chapter IV, in order to evaluate the final controller designs for implementation as CGT/PI/KF controllers requires that additional software be written. These results provide the basis for final design iteration and demonstrate the ultimate model-following performance which may be achieved.

6.2 Using the CGTPIF Design Program

6.2.1 Introduction.

Many different controller designs have been developed using the CGTPIF design computer program. The program has proven to be readily applicable to these different problems despite the many and varied dimensions needed to define the dynamics models relevant to each. The preservation of design information both between separate executions and within a single execution of the program substantially aids efficient design iteration. Also, the information available directly at the user's interactive computer terminal during program execution has been found appropriate and adequate for intelligent selection of design parameters necessary to pursue designs to successful conclusion.

This chapter discusses results obtained by using CGTPIF for specific design problems, but does not detail the program entries these designs entailed. An example of the input and output for such a design is included in Appendix C.

6.2.2 Operating Considerations.

A few simple considerations of the operation of the design program particularly impact its ease of application to specific design problems. These and other more specific considerations are discussed more fully in context in the "Programmer's Manual" (Appendix A) and the "User's Manual" (Appendix B), and the reader is encouraged to refer to those manuals before attempting to use the program. The following should be noted:

1. The design model may be entered only once during a single program execution. Thus, in order to modify the design model, it is necessary to re-execute the program.

2. The truth and command models may be redefined as often as desired during a single program execution. This facilitates evaluation of performance when subject to design model errors, and allows design iteration involving changes to the command model.

3. The "SAVE" and "DATA" files should be used to preserve the design, truth, and command models between distinct program executions. The two files are different according to their local file names: the "SAVE" file is a write-only file created by CGTPIF during program execution, the "DATA" file is a read-only file which the user provides (if desired) by assigning a previously created "SAVE" file the local file name "DATA." Manipulation of local file names does not occur during program execution.

4. Models may be read from the "DATA" file repeatedly during program execution. Thus, if specific elements of the matrices defining the command and/or truth models are to be modified during program execution, it is most convenient to retain nominal representations on a "DATA"

file. The model may then be read and modified readily during subsequent program execution.

5. Changes in model dimensions require that all relevant defining matrices be entered anew. Such change cannot be achieved by modifying existing models of different dimension as through augmentation. Note also that there is some interdependence among the model definitions implicit in the specific dimensions of each. Thus, intended changes in dimensions for a single model may require changes in definition for another dynamics model also. Examples of such implied constraints are equality in number of outputs for the design and command models, and of inputs and of measurements for the design and truth models. Also, because of the need to define the correspondence between the states of the design and truth models, changes in the design model may entail changes in the truth model's $\underline{T}_{DT}$ and $\underline{T}_{NT}$ matrices (equations 3-21a,b).

6. Open-loop CGT designs can only be pursued in a given execution of the program if a PI regulator design has not previously been accomplished and PI gains have not been read from the "DATA" file. The program logic assumes the usual design will be of a CGT/PI controller and pursues such a design automatically in the CGT design path whenever PI gains exist within program storage.

7. Any of the three design paths (PI regulator, CGT controller, or Kalman filter) may be executed in any

order and as often as desired during a single program execution. Results from each design path are preserved independently throughout execution.

8. In addition to the output available at the user's terminal, the same and much additional information is output to the file named "LIST." Following execution, it is good practice to route it to a line printer for listing.

9. Plots of system time-response at the user terminal are the most useful evaluation tool. Each plot requires about 2 minutes to be printed. In evaluating CGT controller response, it is most convenient to include corresponding system and model outputs in the same plot for direct comparison.

10. If it is desired to include results for all of the initial controller sample times, a value for plot duration that is less than 50 times the controller sample period should be specified.

11. Regardless of the plot duration specified, the plot will include 50 time samples and will therefore require a fixed amount of time to be printed. These samples are uniformly distributed in time over the entire duration. The time duration of the response may be adjusted by CGTPIF automatically to achieve 50 evenly spaced time samples coincident with controller sample times.

12. In order to bypass the time-response evaluation, one need only specify that no plots are wished. For the

CGT evaluation, specifying a zero index for the command input also bypasses the time-response evaluation.

6.2.3 Interpreting Plots to the Terminal. The plots of system response produced by CGTPIF are of the "line-printer" type. As printed at the terminal, the time axis runs vertically down the page with the initial time at the top-left margin and time points printed for each sample down the left margin. The dependent variables are plotted in the horizontal sense from left-to-right for increasing magnitude. Each variable plotted is marked by a distinct number (1 through 5) at each sample time. In the event that two variables occupy the same location in the plot field at a sample time, only the plot symbol of largest value will be marked at the point in question (e.g., if the variables represented by symbols 1 and 3 both quantize to the same print location at time t_N, the print position will be filled with symbol 3). Such coincidences of position can be inferred from the behavior of the variable with missing symbol at proximate time samples. Note that a special case of this is when two or more variables to be plotted actually represent the same variable (as may occur, for example, when an output for a system is simply a state and both are plotted); in such a case only the corresponding plot symbol of largest value will appear in the plot field.

The horizontal width of the plot field is 50 print positions. The scale of each plot is printed along the bottom margin of the plot and includes the values of the grid at each multiple of 10 print positions. In the usual case, each variable is plotted on its own unique scale and thus a scale range for each is given corresponding to the plot symbol used. However, to facilitate comparisons of system and model output responses, any plot which includes a model output variable applies a single scale encompassing the full range of all variables.

The plots included in this chapter are those produced by CGTPIF at the user terminal in the course of achieving the designs discussed. Although the plots originally included identifying titles, these have been removed. As presented, the plots are rotated 180° from their usual orientation. By rotating the bound thesis 90° clockwise the independent axis (time) is horizontal and the dependent axis is vertical; the initial time is at lower-left and the minimum value of each scale range is along the bottom grid line; scale identification is along the right vertical margin.

6.3 Simple Design Example

6.3.1 Design Model. The design model for the simple design example includes a system state differential equation in modified canonical form (Ref 31)

$$\dot{\underline{x}} = \begin{bmatrix} -2. & 5. \\ -5. & -2. \end{bmatrix} \underline{x} + \begin{bmatrix} 3. & -1. \\ 1. & 5. \end{bmatrix} \underline{n}_d + \begin{bmatrix} 0. \\ 1. \end{bmatrix} u \tag{6-3a}$$

A disturbance state differential equation

$$\dot{\underline{n}}_d = \begin{bmatrix} 0. & 0. \\ 0. & 0. \end{bmatrix} \underline{n}_d \quad , \tag{6-3b}$$

and an output equation

$$y = [1. \quad 0.]\ \underline{x} \tag{6-3c}$$

Note that the disturbances are simply constants. Responses for the controlled system for disturbances acting or not acting on the system are run by setting initial conditions on the disturbance states as

$$\underline{n}_d(0) = \begin{bmatrix} 0. \\ 0. \end{bmatrix} \tag{6-4a}$$

or as

$$\underline{n}_d(0) = \begin{bmatrix} -1. \\ 2. \end{bmatrix} \tag{6-4b}$$

respectively. Thus the disturbances are of significant magnitude for the responses run "with disturbances."

The transfer function relating the output to the input is,

$$\frac{y}{u} = \frac{5}{s^2 + 4s + 29} \tag{6-5}$$

which has a natural frequency of about 5.4 radians per second and a damping ratio of about 0.37. This model has a frequency and damping typical of aircraft "short-period" dynamics.

6.3.2 Truth Model. The nominal truth model represents the same system dynamics as the design model, incorporating the disturbance states into the truth model state vector:

$$\dot{\underline{x}}_t = \begin{bmatrix} -2. & 5. & 3. & -1. \\ -5. & -2. & 1. & 5. \\ 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. \end{bmatrix} \underline{x}_t + \begin{bmatrix} 0. \\ 1. \\ 0. \\ 0. \end{bmatrix} u_t \qquad (6\text{-}6a)$$

The matrices relating the design and truth model states are, for the design model's system states

$$\underline{T}_{DT} = \begin{bmatrix} 1. & 0. & 0. & 0. \\ 0. & 1. & 0. & 0. \end{bmatrix} \qquad (6\text{-}6b)$$

and for the design model disturbance states

$$\underline{T}_{NT} = \begin{bmatrix} 0. & 0. & 1. & 0. \\ 0. & 0. & 0. & 1. \end{bmatrix} \qquad (6\text{-}6c)$$

An alternate dynamics matrix ($\underline{A}_t$) for the truth model is used to evaluate the degradation in performance due to misrepresentation of system dynamics in the design model:

$$\underline{A}_t' = \begin{bmatrix} -1. & 5. & 3. & -1. \\ -5. & -1. & 1. & 5. \\ 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. \end{bmatrix} \tag{6-7}$$

This corresponds to a transfer function for system output to input of

$$\frac{\underline{y}}{u} = \frac{5}{s^2 + 2s + 26} \tag{6-8}$$

which has a natural frequency of about 5.1 radians per second and a damping ratio of about 0.20.

6.3.3 Command Models.

6.3.3.1 First-Order Command Model CM(01).

For a simple transfer function relating the desired output response to a command input given by

$$\frac{y_m}{u_m} = \frac{5}{s+5} \tag{6-9}$$

an appropriate state model is readily determined and is represented as

$$\dot{x}_m = -5x_m + u_m \tag{6-10a}$$

with output equation

$$y_m = 5x_m \tag{6-10b}$$

For this model, the output response is a simple exponential with a 0.2 second time-constant. The command model defined by equations (6-10a,b) is hereafter referred to as "CM(O1)."

6.3.3.2 Second-Order Command Models CM(O2C), CM(O2C)´. A second-order command model with complex poles of

$$s_{1,2} = -5. \pm j5. \tag{6-11}$$

and transfer function relating model output response to command input of

$$\frac{y_m}{u_m} = \frac{50}{s^2 + 10s + 50} \tag{6-12}$$

is chosen. This gives a good representation of desirable short-period dynamics, having a natural frequency of about 7.1 radians per second and damping ratio of 0.707. State differential equations are obtained corresponding to the modified canonical form and standard controllable form representations. The command models for each form are hereafter referred to as "CM(O2C)" and "CM(O2C)´" respectively.

The state differential equation for CM(O2C) is,

$$\underline{\dot{x}}_m = \begin{bmatrix} -5. & 5. \\ -5. & -5. \end{bmatrix} \underline{x}_m + \begin{bmatrix} 0. \\ 1. \end{bmatrix} u_m \tag{6-13a}$$

and the output equation is,

$$y_m = [10. \quad 0.]\ \underline{x}_m \tag{6-13b}$$

The standard controllable form of CM(O2C)′ is simply

$$\underline{\dot{x}}'_m = \begin{bmatrix} 0. & 1. \\ -50. & -10. \end{bmatrix} \underline{x}'_m + \begin{bmatrix} 0. \\ 1. \end{bmatrix} u_m \tag{6-14a}$$

with output equation

$$y_m = [50. \quad 0.]\ \underline{x}'_m \tag{6-14b}$$

6.3.4 *Design for CM(O1)*. An open-loop CGT controller for the system of equations (6-3a,b,c) using command model CM(O1) was designed first. The values of the feedforward gains are (from equations (3-49) and (3-52)),

$$\underline{K}_{x_m} = 32.33 \tag{6-15a}$$

$$\underline{K}_{x_u} = -.6657 \tag{6-15b}$$

and

$$\underline{K}_{x_n} = [-2.2 \quad -4.6] \tag{6-15c}$$

Figure 6-1 shows the system output response to a unit step input to the command model with no disturbances acting on the system, and using the nominal truth model. Plot symbol 1 is the system output and plot symbol 2 is the command model output. Note that the system response is slow and oscillatory with a peak overshoot of about 14%. Other time-response runs (not shown here) verified that model-following was achieved in steady-state. Clearly for this controller a closed-loop CGT/PI is appropriate even for the case of no disturbances and perfect system modeling.

A PI regulator was designed, employing quadratic weights of 100. on the output, 500. on state 2, and 1. on the control input magnitudes and rates. Equal weights for all inputs was arbitrary initially and a value of unity was used as a scale basis for the quadratic weights. Weights on the output and state 2 approximately provide output and output rate weighting, and the values selected are simply relative to those for the inputs. The resulting feedback gain matrix for the simple regulator is (from equation (3-77))

$$\underline{G}_c^* = [-0.1028 \quad 0.6440 \mid 0.4352] \qquad (6\text{-}16)$$

and the PI regulator gains are (from equation (3-90))

$$\underline{K}_x = [16.54 \quad 21.41] \qquad (6\text{-}17a)$$

and $$\underline{K}_z = 2.679 \qquad (6\text{-}17b)$$

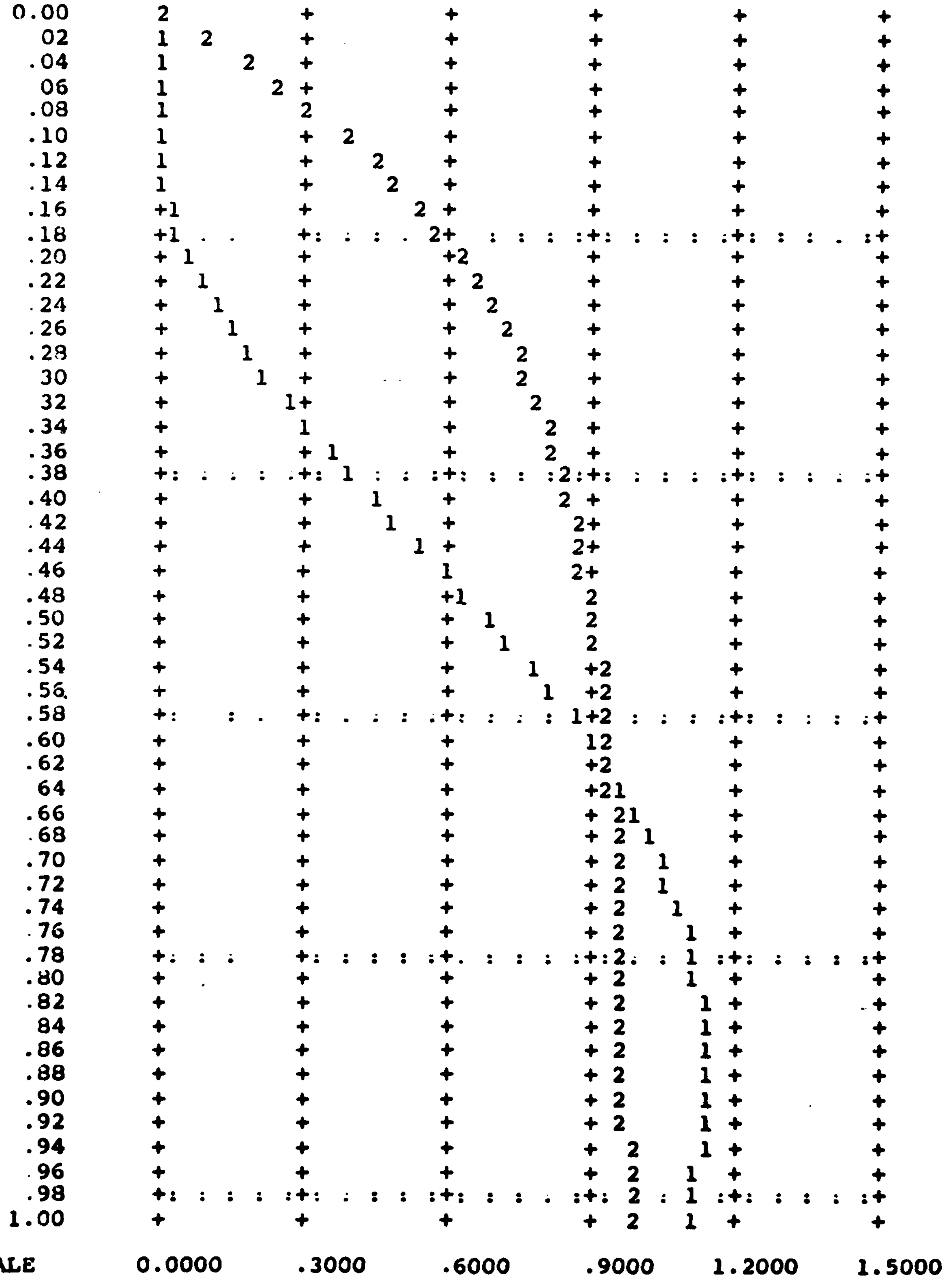

Fig. 6-1. Open-Loop CGT for CM(01), (No Disturbances, Nominal Truth Model)

Note that the PI gains are much larger than the gains of the simple regulator. Figure 6-2 shows the closed-loop response for initial conditions of 1. on both system states and zero on both disturbance states for the nominal truth model. Plot symbols of 1 and 2 signify the system states 1 and 2, respectively (the system output is state 1). The response achieved is rapid and well-damped.

The corresponding CGT/PI controller feedforward gains then are

$$\underline{K}_{x_m} = 50.91 \tag{6-18a}$$

$$\underline{K}_{x_u} = 20.72 \tag{6-18b}$$

and

$$\underline{K}_{x_n} = [-15.05 \quad -.3172] \tag{6-18c}$$

As can be seen from equations (6-2a,b,c), the feedforward gains depend in part on the feedback gain $\underline{K}_x$; in general, increases in quadratic weights result in increased values in $\underline{K}_x$ and thence in the feedforward gains.

The time response for a command input step of unity, with no disturbance and nominal truth model is shown in Figure 6-3. Plot symbols 1 and 2 are the system and command model outputs, respectively. The resulting response demonstrates the benefit achievable by employing inner-loop PI regulation to null errors between the actual system response and the response commanded by the CGT feedforward control laws. Although the model-following is

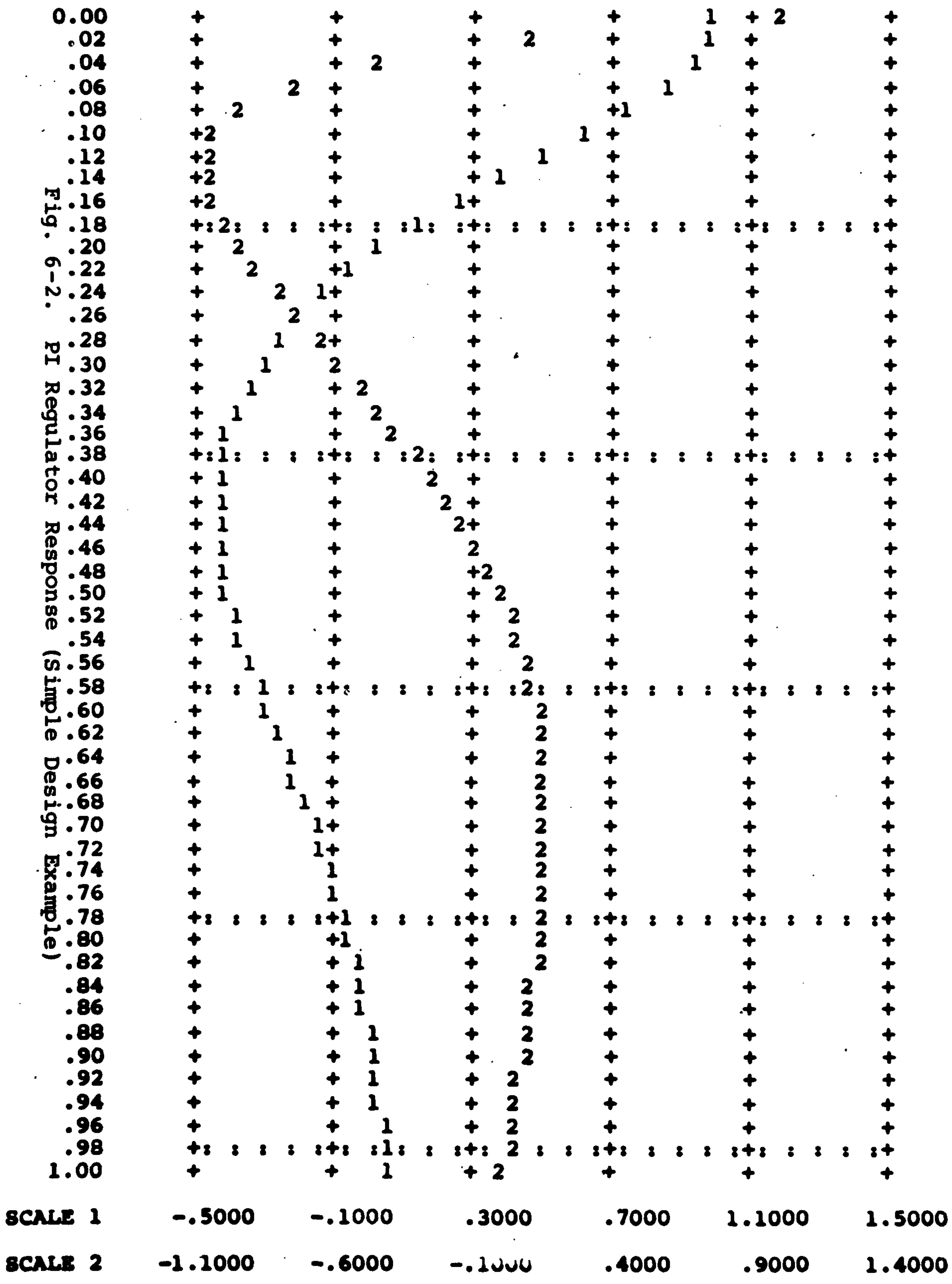

Fig. 6-2. PI Regulator Response (Simple Design Example)

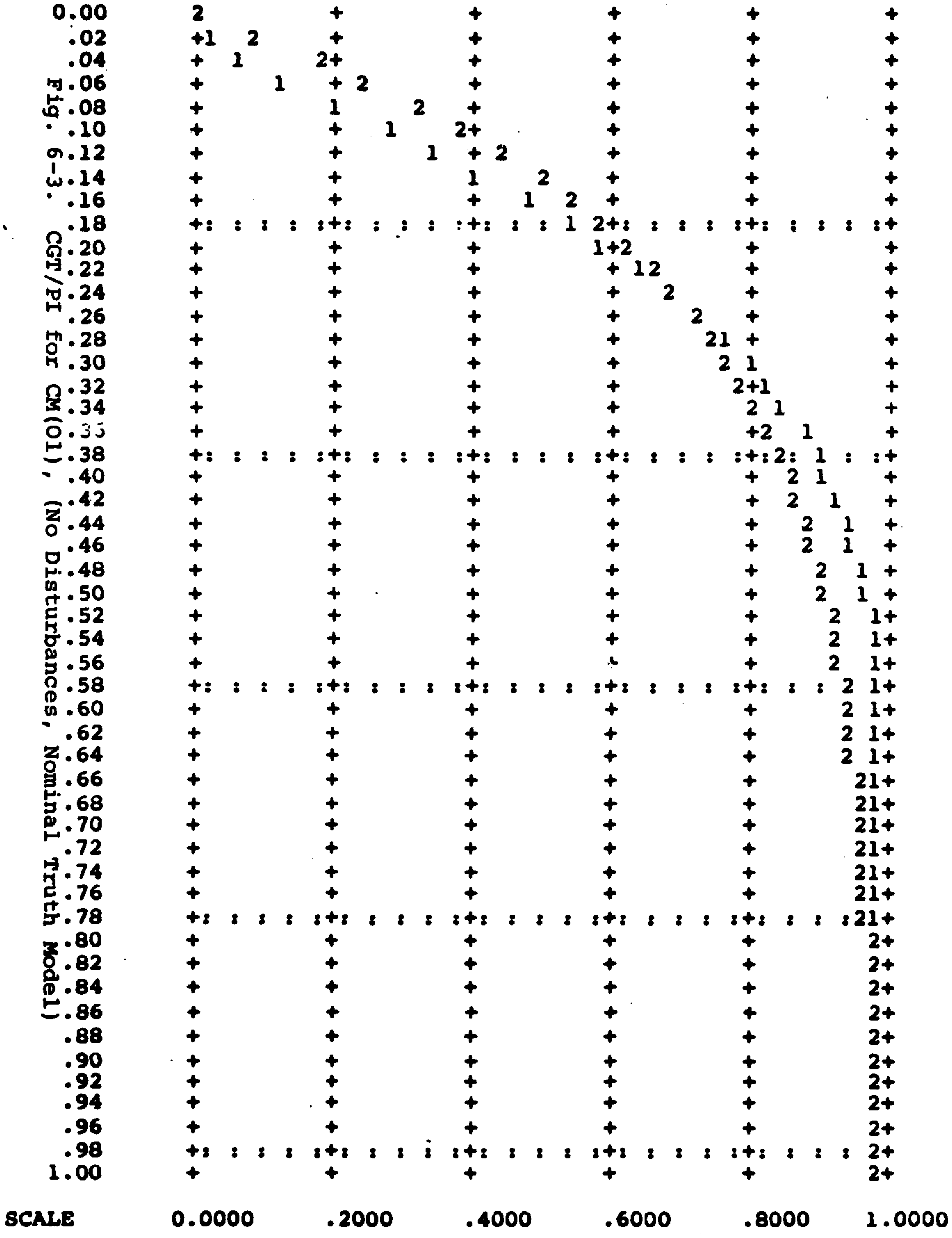

Fig. 6-3. CGT/PI for CM(O1), (No Disturbances, Nominal Truth Model)

not perfect, the system output response is clearly a good representation of the desired dynamics.

6.3.5 Design for CM(O2C) and CM(O2C)´. Results obtained using CM(O2C) are discussed in detail for open- and closed-loop CGT designs with and without disturbances and for nominal and alternate truth models. As expected, different representations of given command model dynamics yielded identical controller performance--only the gains on command model states differed. Thus, gains for designs based upon CM(O2C) and CM(O2C)´ are presented together, but only plots for CM(O2C) designs are shown.

Gains for an open-loop CGT based upon CM(O2C) are

$$\underline{K}_{x_m} = [20. \quad -56.13] \tag{6-19a}$$

$$\underline{K}_{x_u} = 9.413 \tag{6-19b}$$

and

$$\underline{K}_{x_n} = [-2.2 \quad -4.6] \tag{6-19c}$$

For the design based upon CM(O2C)´, gains $\underline{K}_{x_u}$ and $\underline{K}_{x_n}$ are unchanged while the model state gains become

$$\underline{K}'_{x_m} = [-180.6 \quad -56.13] \tag{6-19a´}$$

Note that the feedforward gain for the command model CM(O2C)´ is the product of the gain of equation (6-19a) for CM(O2C) and the transformation matrix relating the two state variable sets:

$$\underline{x}_m = \underline{T}\underline{x}_m' \tag{6-20a}$$

in which $\underline{x}_m$ and $\underline{x}_m'$ are the state vectors of CM(O2C) and CM(O2C)´, respectively and the transformation matrix is

$$\underline{T} = \begin{bmatrix} 5. & 0. \\ 5. & 1. \end{bmatrix} \tag{6-20b}$$

and is obtained readily since CM(O2C)´ is in standard controllable form (Ref 19). Relating the feedforward gains for the two state descriptions leads to

$$\underline{K}_{x_m} \underline{x}_m = \underline{K}_{x_m} (\underline{T}\underline{x}_m') \tag{6-20c}$$

and finally gives the simple result that

$$\underline{K}_{x_m}' = \underline{K}_{x_m} \underline{T} \tag{6-20d}$$

which is verified for the values of $\underline{K}_{x_m}'$, $\underline{K}_{x_m}$, and $\underline{T}$ given by equations (6-19a´), (6-19a), and (6-20b) above.

Figures 6-4 and 6-5 show the response of the system using the open-loop CGT controller for the nominal truth model (equation (6-6)), without and with disturbances, respectively. The case of no disturbance but alternate truth model (equation (6-7)) is given in Figure 6-6. A unit step on the command input is applied in all three responses. Plot symbol 1 is the system output and symbol 2 is the command model output.

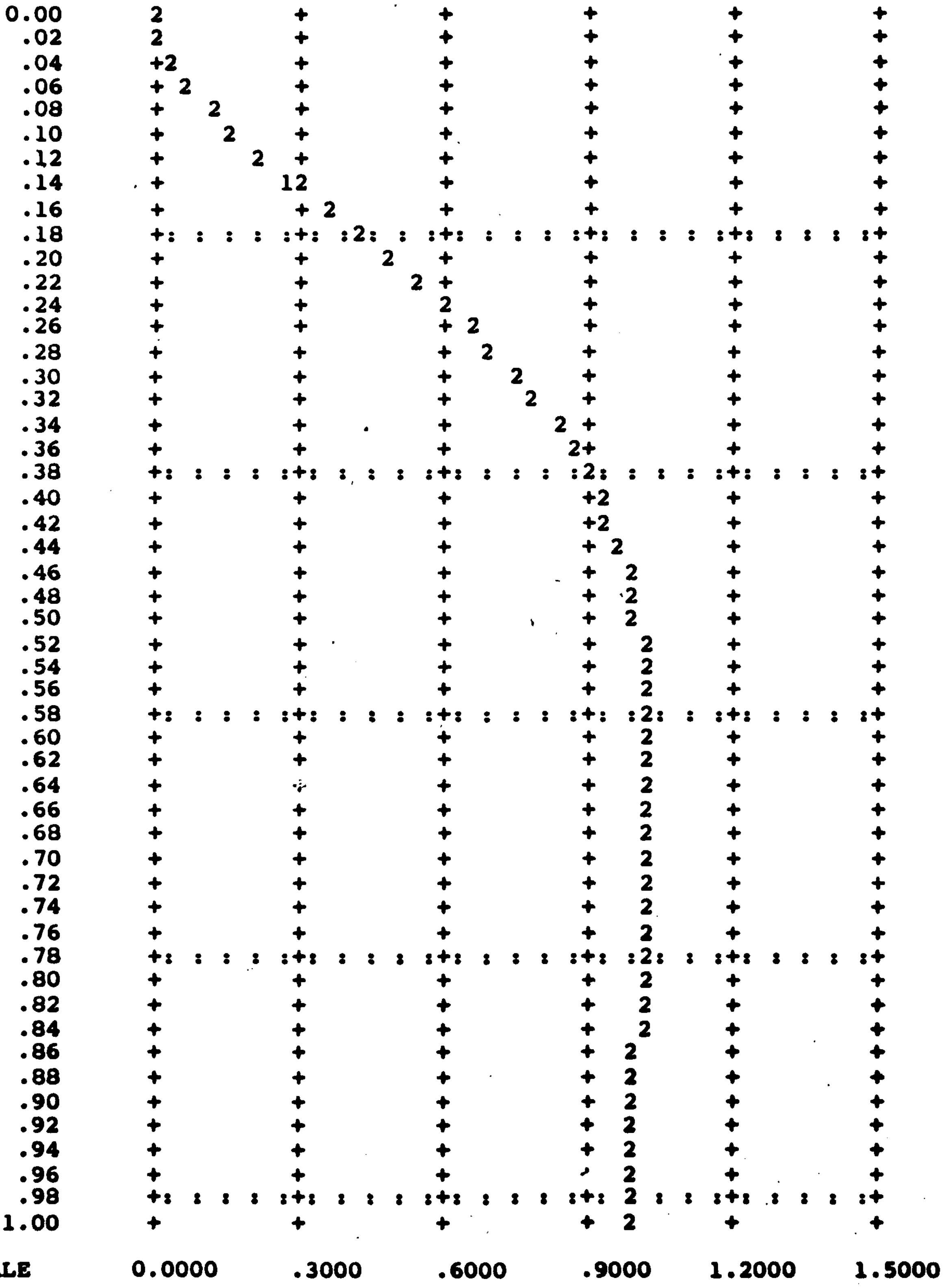

Fig. 6-4. Open-Loop CGT for CM(O2C), (No Disturbances, Nominal Truth Model)

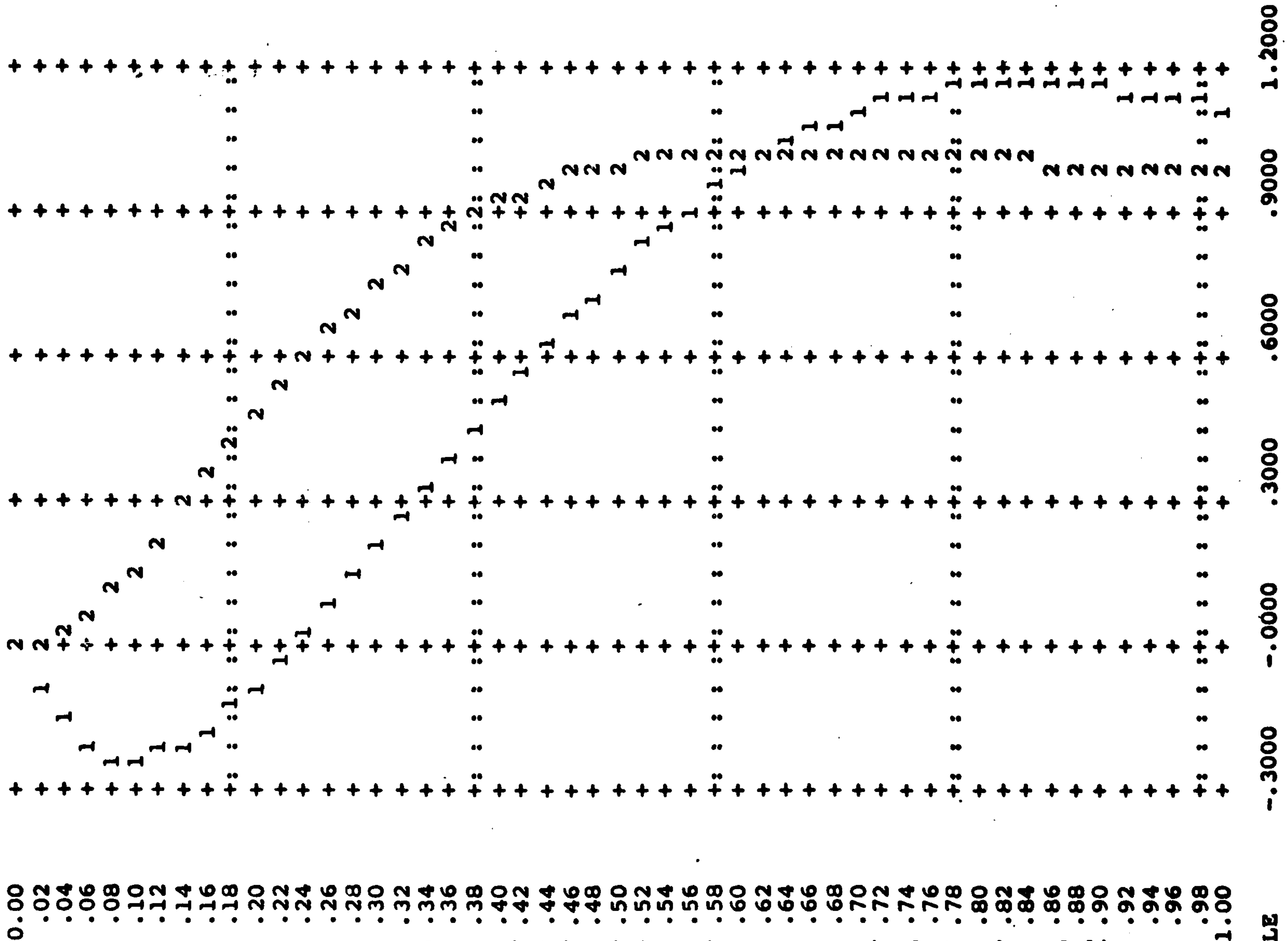

Fig. 6-5. Open-Loop CGT for CM(O2C), (Disturbances, Nominal Truth Model)

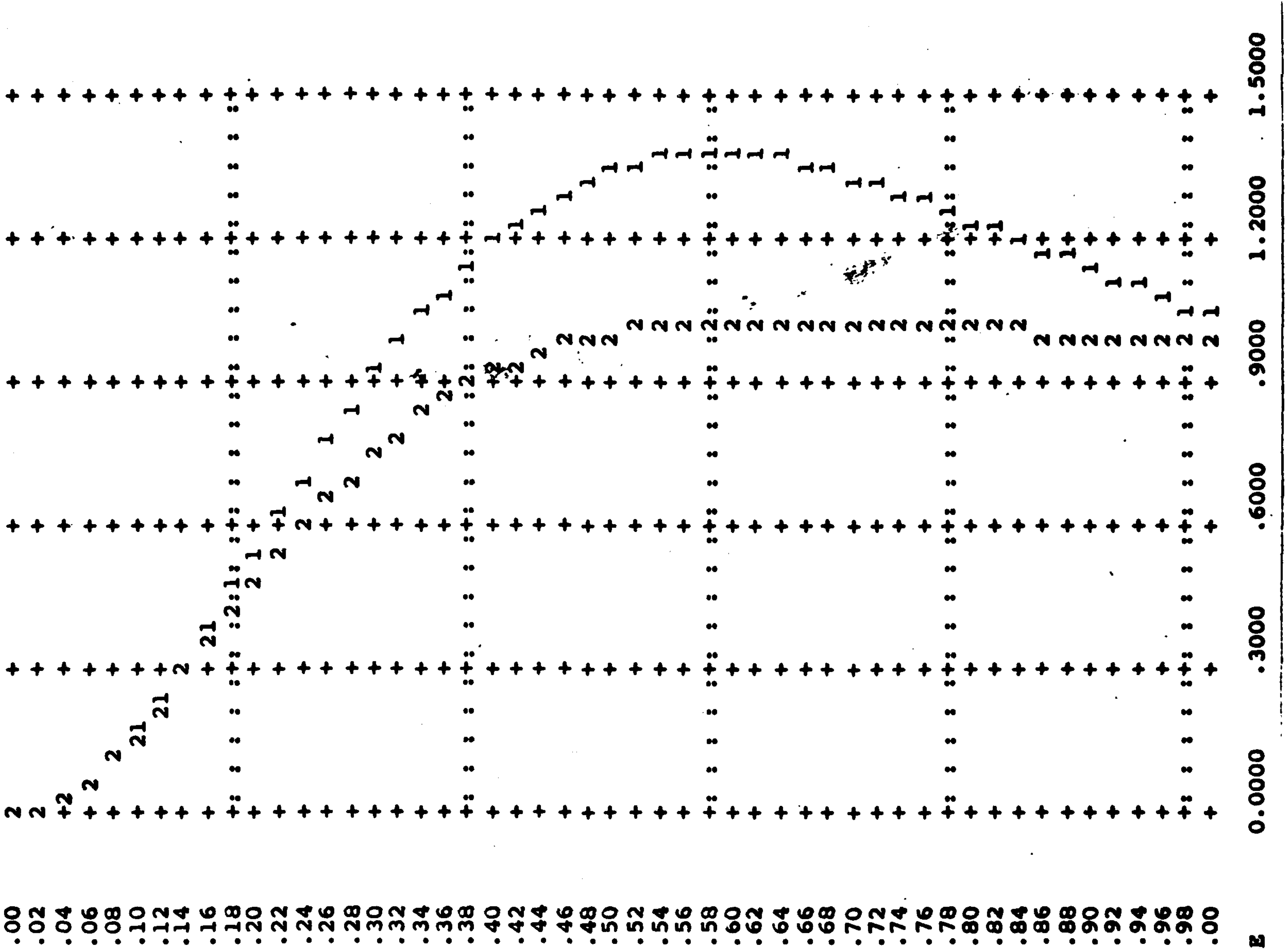

Fig. 6-6. Open-Loop CGT for CM(O2C), (No Disturbances, Alternate Truth Model)

As seen in Figure 6-4, for this design model-command model pair, the open-loop CGT controller achieves essentially perfect model-following when no disturbances affect the dynamics and the design model is an exact representation of true system dynamics. Figures 6-5 and 6-6 demonstrate the deterioration in transient performance when disturbances are present or design model errors exist. However, the modeled disturbances are actually successfully overcome by the open-loop CGT controller in steady-state; the bounds on system output response are -0.25 and +1.15. In the time-response of Figure 6-6 the actual response reflects the lower damping of the alternate truth model and has an overshoot of about 38%.

In direct contrast to these open-loop results, Figures 6-7, 6-8, and 6-9 show the corresponding cases employing a closed-loop CGT/PI controller. The same PI regulator gains determined in Section 6.3.4 above are used in the inner-loop. For the design based upon CM(O2C), the feedforward gains are determined as

$$\underline{K}_{x_m} = [56.76 \quad 157.7] \tag{6-21a}$$

$$\underline{K}_{x_u} = 9.456 \tag{6-21b}$$

and

$$\underline{K}_{x_n} = [-15.05 \quad -.3172] \tag{6-21c}$$

The gains for the design based upon CM(O2C)´ are identical to these for $\underline{K}_{x_u}$ and $\underline{K}_{x_n}$, while the feedforward gains on the command model states become

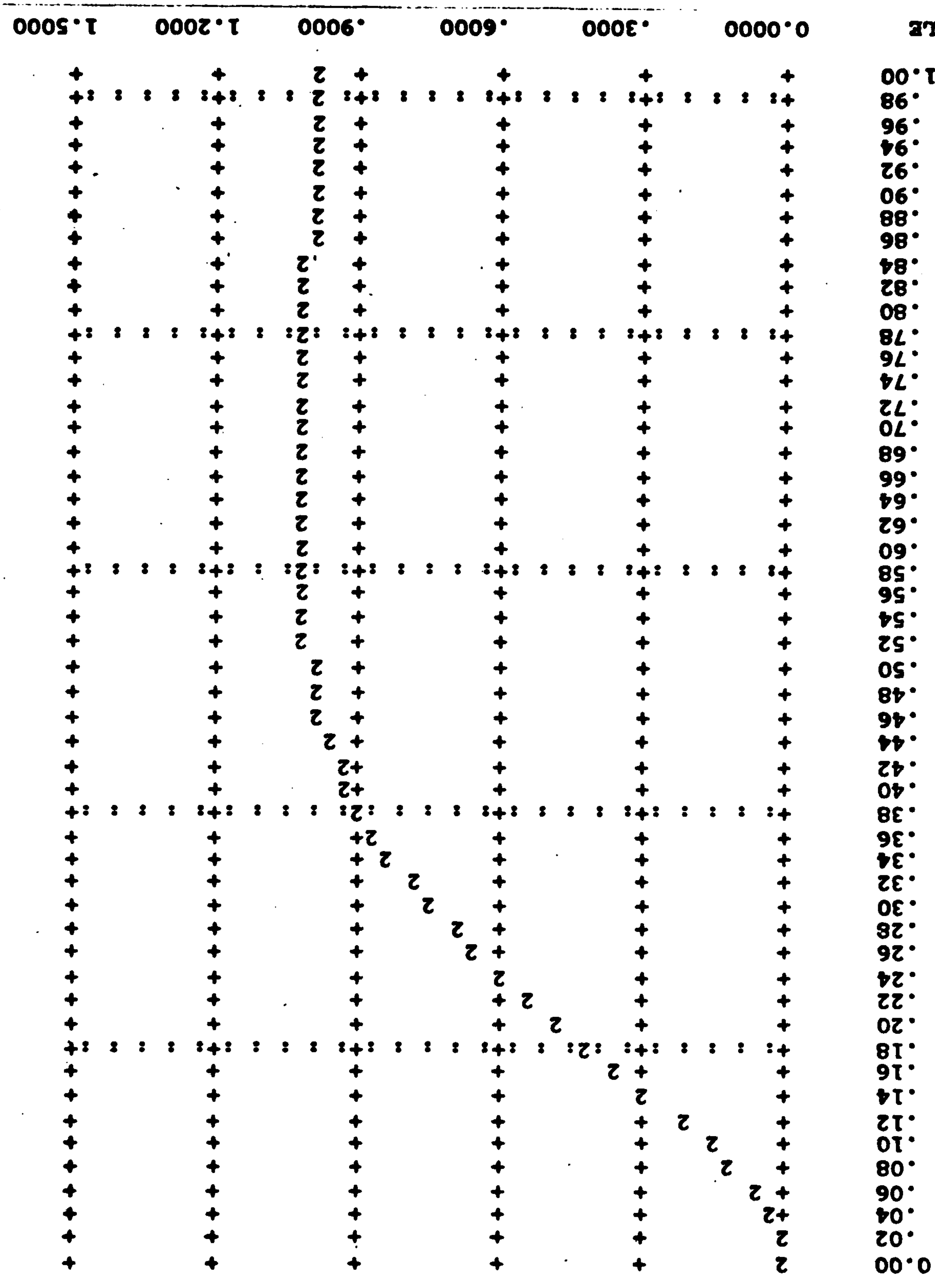

Fig. 6-7. CGT/PI for CM(O2C), (No Disturbance, Nominal Truth Model)

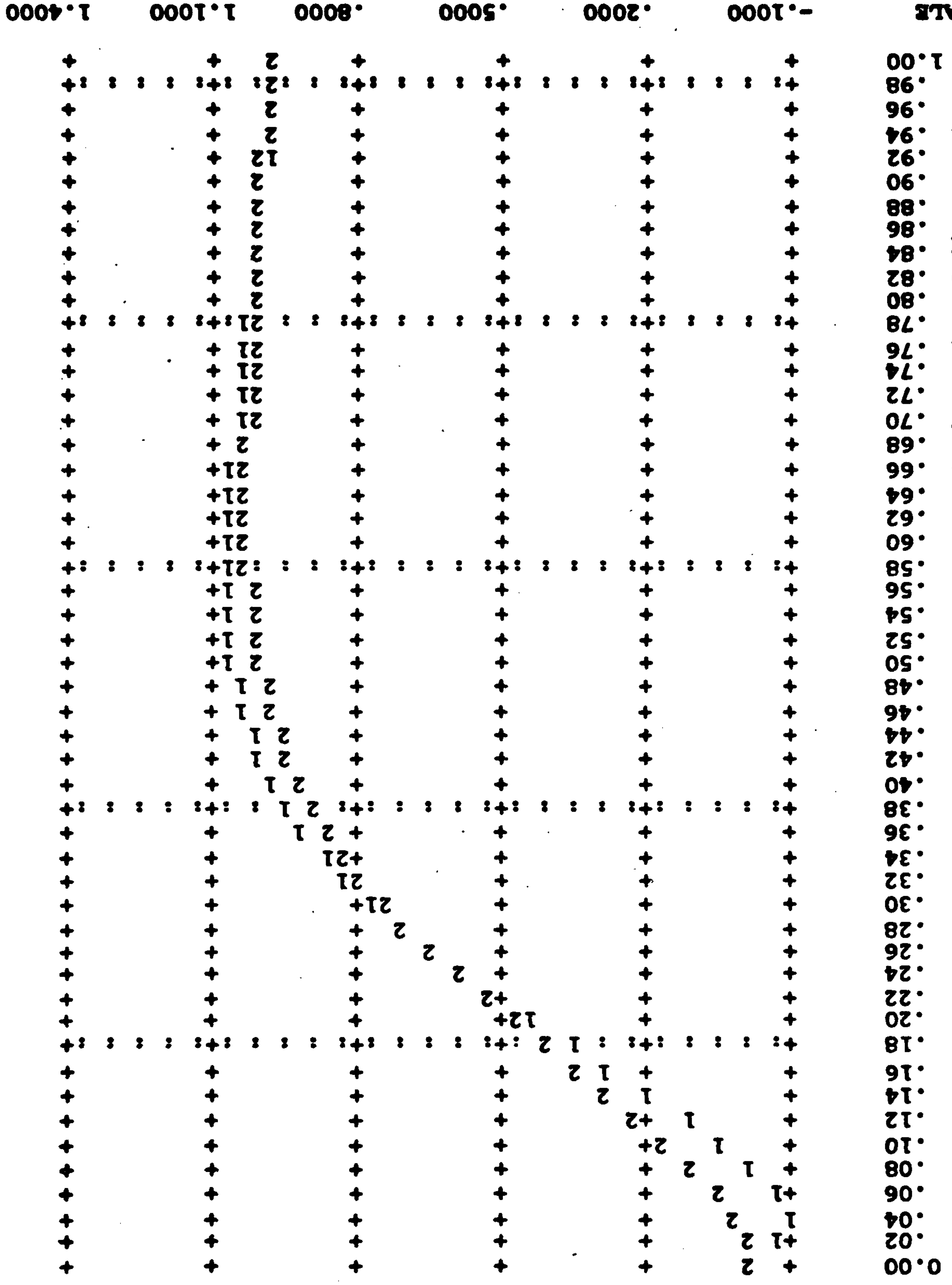

Fig. 6-8. CGT/PI for CM(O2C), (Disturbances, Nominal Truth Model)

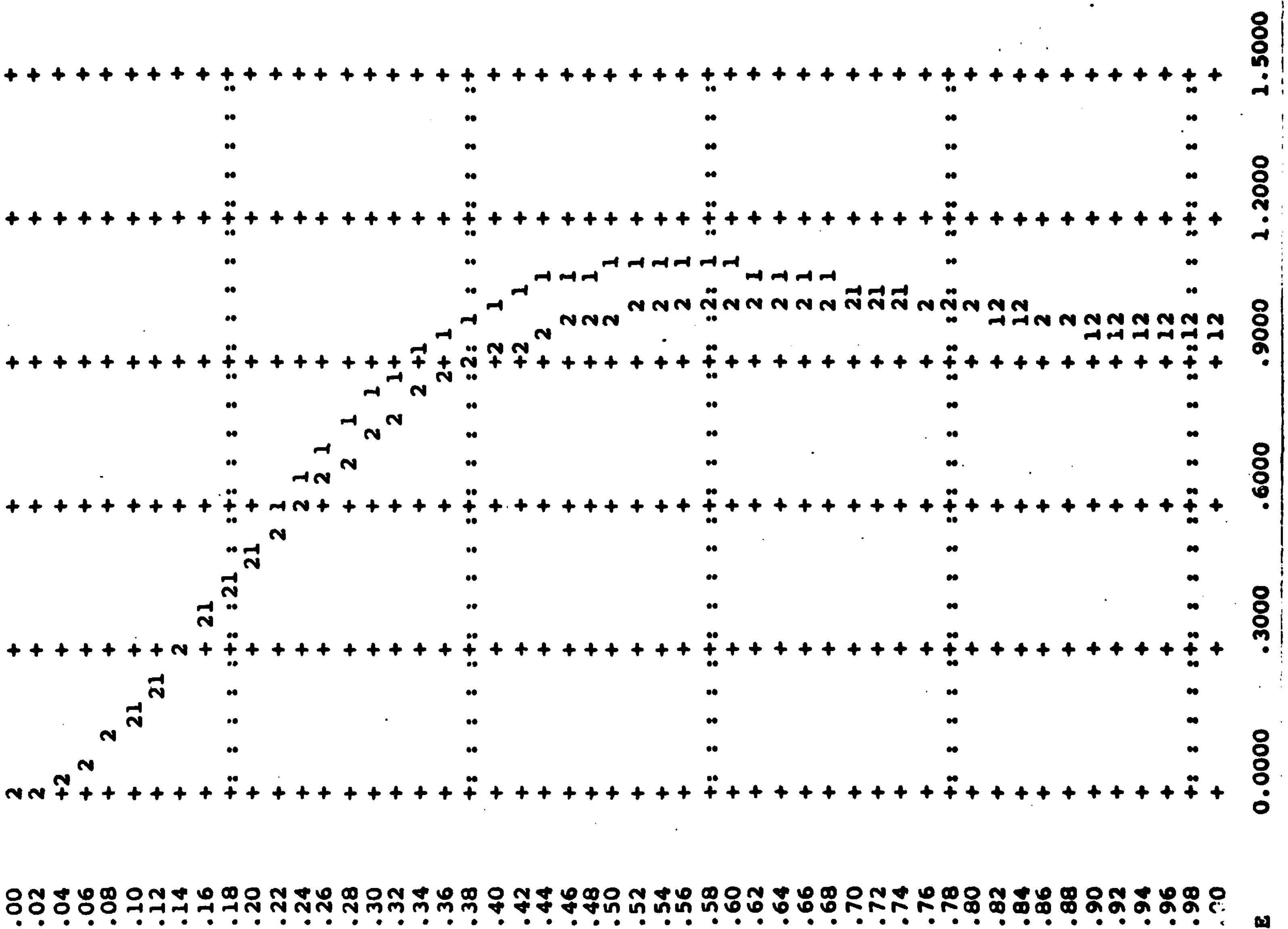

Fig. 6-9. CGT/PI for CM(O2C), (No Disturbances, Alternate Truth Model)

$$\underline{K}'_{x_m} = [1072. \quad 157.7] \qquad (6\text{-}22)$$

Note that these closed-loop values for $\underline{K}'_{x_m}$ and $\underline{K}_{x_m}$ are related as before through equation (6-20d).

The plots of Figure 6-7 show that the system and command model outputs are coincident (within the resolution of the printer plots) using the CGT/PI controller. More significantly, the responses of Figure 6-8 and 6-9 demonstrate considerable improvement in the transient response of the controlled system in the face of non-zero modeled disturbances and of modeling errors, respectively. With disturbances the response is within the bounds of -0.1 to +1.07 and the steady-state model-following is quickly achieved, while even during the transient rise portion the matching is quite good. Similarly for design model dynamics errors, the transient behavior is much improved, an overshoot of only about 11% occurs, and the steady-state model-following is quickly achieved.

6.3.6 Discussion of Results for Simple Design Example. Comparing the various feedforward gain matrices given earlier in this section, it can be seen that the gains applied to the disturbance states are independent of the command model. Also, for given command model dynamics, the gains on the model states vary according to the specific state representation employed, but the gains on command model inputs do not, as seen in Section 6.3.5.

Designs for CGT controllers based on command models CM(O2C) and CM(O2C)´ demonstrated the invariance of controlled system behavior with alternative state representations of specified dynamics. Thus, particularly for SISO systems, desirable dynamic behavior may be formulated readily into command model state representations, as through translation from transfer function representation to a state representation in standard controllable form. Design iteration then primarily involves the inner-loop PI regulator or in some cases may entail modifications in the proposed dynamics but not in the representations achieving such dynamics.

As expected, the closed-loop CGT/PI controller shows significant advantages over an open-loop CGT controller. Transient response is improved, modeling errors are more readily overcome, steady-state exact model-following is obtained more quickly, and rejection of modeled disturbances in the transient phase of response is superior. Additional tests aimed at exploring the qualities provided specifically by the PI regulator (as opposed to standard regulator designs) could be performed. Since the designs employed in this study assumed use of PI regulation, such additional tests have not been examined. However, for example, it is clear that the CGT/PI controller response when subject to unmodeled disturbances would be superior (would achieve zero steady-state error) to either the open-loop CGT or closed-loop CGT employing simple inner-loop regulation.

6.4 Aircraft Flight Control Design

6.4.1 Introduction. The designs discussed in this section are based upon the dynamics corresponding to the Advanced Fighter Technology Integration aircraft, an F-16 aircraft (AFTI/F-16) specially modified for the purpose of advanced controls research. Two distinct controllers are designed based upon longitudinal dynamics about straight and level flight at Mach 0.8 and an altitude of 10000. feet. This particular flight condition was chosen as representative of an operating point within the usual air-to-air combat flight regime. Since the AFTI/F-16 is unstable in pitch in this regime, open-loop CGT designs are infeasible. Therefore, only CGT/PI designs are pursued.

One of the concepts the AFTI/F-16 aircraft is to demonstrate is the feasibility and usefulness of unconventional decoupled pitch modes of motion compared with conventional pitch control. Decoupled pitch control is possible because of two independently controlled longitudinal control surfaces: horizontal tail and trailing-edge flap. The designs discussed here are for a so-called pitch-pointing controller and a pitch rate controller.

The pitch-pointing controller is in the class of unconventional, decoupling controllers. The objective is to achieve direct control of aircraft pitch attitude while constraining the flight-path angle to remain constant. Thus the aircraft may be pointed in pitch through small angles without changing the flight trajectory. Such control is useful, for example, in air-to-air target tracking to null gun-aiming error briefly when the trajectory matching is good but small errors in the aim-point persist in the pitch plane. The pitch-pointing control mode may be practically achieved through appropriate control input command to the two independent control surfaces. However, such designs are difficult to achieve using conventional control design techniques since the problem inherently requires effective multi-input-multi-output (MIMO) design methodology. The CGT/PI controller design technique is readily applicable to such MIMO decoupling design problems, systematically determining appropriate crossfeeds while assuring closed-loop stability.

The pitch rate controller is conventional in aircraft flight control, except that for this case appropriate utilization of both control surfaces is problematical when using conventional control design methods. For the CGT/PI control design technique, such conventional control objectives represent a special case in which model-following is desired for fewer outputs than the number of available control inputs. The CGT/PI design described subsequently abides by the requirement (imposed in Sections 3.2.1 and 3.3.1) that the numbers of outputs and inputs of the design model and outputs of the command model all be equal, but treats only one output of each model as a non-zero combination of states. Although the resulting augmented matrix inverse $\underline{\Pi}$ of equation (3-47) does not exist, the matrix pseudo-inverse is computed instead (Ref 32). The final CGT/PI controller achieves the desired model-following in the single output channel specified. For this case in which more controls are available than the number of outputs to have specified dynamics, the pseudo-inverse provides an exact controller solution (Ref 32). However, in the opposite case of fewer inputs than outputs, the use of the pseudo-inverse would give an approximate solution only.

Data describing the AFTI/F-16 was obtained from References 3 and 18. Reference 18 gives the dimensional stability derivatives in the aircraft body-axis coordinate frame in angular units of radians, but does not include data for the trailing-edge flap. The corresponding derivatives for the trailing-edge flap were computed from the data of Reference 3 which gives the data in the stability-axis frame and in angular units of degrees. The state-space representations employed in describing the aircraft longitudinal dynamics are given below. Reference 30 details the derivation of the linear perturbation model of aircraft longitudinal dynamics while the model of clear-air turbulence is obtained from Reference 21.

Various models are used in specific design examples presented below. After describing the full truth model used for the longitudinal dynamics, several simplifications are proposed. These simplifications are used to demonstrate the effects of parameter variation and of unmodeled and modeled control surface actuator lags on the CGT/PI time-response. Finally, one of these simplified models is used as the design model for a Kalman filter design. The covariance analysis is with respect to the full truth model.

6.4.2 <u>AFTI/F-16 Truth Model</u>. The full truth model for the AFTI/F-16 aircraft employs four system states, two actuator states, and three turbulence (gust) states. Noise corrupted measurements are assumed available for three of the system states. In addition, one of the measurements includes a gust state. The model is hereafter referred to as AFTI(S4,A2,G3), i.e., the model based on 4 system states, 2 actuator states, and 3 gust states. Computations needed to determine the model parameters were formulated in subroutines and included in the final object file of CGTPIF.

The state vector is defined to consist of the following elements:

$$\underline{x}_t = \left[\begin{array}{c} \theta \\ \alpha \\ q \\ u \\ \hdashline \delta_{HT} \\ \delta_{TEF} \\ \hdashline \alpha_g' \\ \alpha_g \\ q_g \end{array}\right] \qquad (6-23)$$

in which the perturbation states are as follows: θ is the aircraft pitch angle, α is the aircraft angle-of-attack, q is the aircraft pitch rate, u is the forward velocity, δ_{HT} is the deflection angle of the horizontal tail, δ_{TEF} is the deflection angle of the trailing-edge flap, α_g' and α_g are angle-of-attack gust states, and q_g is a pitch rate gust state. Since the aircraft state description is obtained in two steps, it is convenient to first describe the actuator and gust models.

Both control surfaces are modeled as driven through first-order lags representing the actuator dynamics. The transfer function for the actuator lag is of the form

$$\frac{\delta_s}{\delta_{s_I}} = \frac{T_s}{s+T_s} \tag{6-24}$$

in which δ_s and δ_{s_I} are the surface deflection and command to the surface input (actuator), respectively. The value of T_s is 20., implying an actuator time-constant of 0.05 seconds for both actuators. The corresponding state model is

$$\dot{\delta}_s = - T_s\delta_s + T_s\delta_{s_I} \tag{6-25}$$

The state equations for the gust model are as given in Reference 21:

$$\dot{\underset{\sim}{\alpha}}'_g = a'_{g_1}\underset{\sim}{\alpha}'_g + \underset{\sim}{\eta}_\alpha \qquad (6\text{-}26a)$$

$$\dot{\underset{\sim}{\alpha}}_g = a'_{g_2}\underset{\sim}{\alpha}'_g + a_g\underset{\sim}{\alpha}_g + a_n\underset{\sim}{\eta}_\alpha \qquad (6\text{-}26b)$$

$$\dot{\underset{\sim}{q}}_g = a_{g_d}\dot{\underset{\sim}{\alpha}}_g + q_{g_1}\underset{\sim}{q}_g \qquad (6\text{-}27c)$$

in which there is only one noise source($\underset{\sim}{\eta}_\alpha$), modeled as a zero-mean white Gaussian noise process with unit variance. The various coefficients are computed as

$$a'_{g_1} = \frac{-V_T}{L_w} \qquad (6\text{-}28a)$$

$$a'_{g_2} = \frac{\sigma_w}{L_w}\,(1.\ -\sqrt{3.})\sqrt{\frac{V_t}{L_w}} \qquad (6\text{-}28b)$$

$$a_g = \frac{-V_T}{L_w} \qquad (6\text{-}28c)$$

$$a_n = \frac{\sigma_w}{V_T}\sqrt{\frac{3V_T}{L_w}} \qquad (6\text{-}28d)$$

$$a_{q_d} = \frac{\pi V_T}{4b} \qquad (6\text{-}28e)$$

$$q_{g_1} = \frac{-\pi V_T}{4b} \qquad (6\text{-}28f)$$

and V_T is the aircraft total velocity in feet per second and b is aircraft wing span in feet. Both σ_w and L_w are determined from specifications given in MIL-SPEC 8785B (Reference 12): σ_w is the root-mean-square intensity for the clear air turbulence (as a function of "Level" and altitude) in feet per second and L_w is the scale length (as a function of altitude) in feet.

It is convenient to define the system dynamics matrix for the truth model in two steps. First, the full matrix will be specified based on zero values for stability derivatives related to the rate of change of angle-of-attack ($\dot{\alpha}$) and neglecting the term involved $\dot{\alpha}_g$ in the pitch rate gust equation. The resulting matrix is then modified to incorporate the non-zero $\dot{\alpha}$ stability derivatives and the $\dot{\alpha}_g$ term in the q_g state equation. Thus, define the initial dynamics matrix as

$$\underline{A}_t' = \left[\begin{array}{cccc|cc|ccc}
0. & 0. & 1. & 0. & 0. & 0. & 0. & 0. & 0. \\
-g(\sin \alpha_0)/U_0 & Z_\alpha & 1.+Z_q & Z_u & Z_{\delta_{HT}} & Z_{\delta_{TEF}} & 0. & Z_\alpha & Z_q \\
0. & M_\alpha & M_q & M_u & M_{\delta_{HT}} & M_{\delta_{TEF}} & 0. & M_\alpha & M_q \\
-g(\cos \alpha_0) & X_\alpha & X_q-W_0 & X_u & X_{\delta_{HT}} & X_{\delta_{TEF}} & 0. & X_\alpha & X_g \\
\hline
0. & 0. & 0. & 0. & -T_s & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & -T_s & 0. & 0. & 0. \\
\hline
0. & 0. & 0. & 0. & 0. & 0. & a_{g_1}' & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & a_{g_2}' & a_g & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & q_g
\end{array}\right] \tag{6-29}$$

A new matrix, $\underline{A}_t$, is next determined by incorporating rows 1, 5, 6, 7, and 8 intact from matrix $\underline{A}_t'$ and modifying the remaining rows as follows:

$$\underline{A}_t(2,I) = \underline{A}'_t(2,I) \;/\; (1.-Z_{\dot{\alpha}}) \qquad (6\text{-}30a)$$

$$\underline{A}_t(3,I) = \underline{A}'_t(3,I) + M_{\dot{\alpha}} \cdot \underline{A}_t(2,I) \qquad (6\text{-}30b)$$

$$\underline{A}_t(4,I) = \underline{A}'_t(4,I) + X_{\dot{\alpha}} \cdot \underline{A}_t(2,I) \qquad (6\text{-}30c)$$

$$\underline{A}_t(9,I) = \underline{A}'_t(9,I) + a_{g_d} \cdot \underline{A}'_t(8,I) \qquad (6\text{-}30d)$$

where the index "I" ranges from 1 to 9 and a_{g_d} is as defined in equation (6-28e). In both equations (6-29) and (6-30) the stability derivatives denoted by "X", "Z", and "M" are related to the forces acting along the aircraft body X- and Z-axes and the pitching moment (about the body Y-axis), respectively. Subscripts indicate the state with respect to which the derivative is determined. The subscripts "δ_{HT}" and "δ_{TEF}" refer to the control surface deflections of horizontal tail and trailing-edge flap. The term "g" is the acceleration of gravity. Finally, α_0, U_0, and W_0 are the trim values of angle-of-attack, forward velocity component along the body X-axis, and the downward velocity component along the body Z-axis. Here, "trim" is the nominal operating point about which linearization is accomplished and is a condition of wings-level, constant velocity, and constant altitude flight at the Mach number and altitude specified for the flight condition (0.8 and 10000. feet respectively).

The values of all the parameters of equations (6-25), (6-28), (6-29), and (6-30) are listed in Table 6-1. The value of σ_w listed is for a "Level 1" turbulence, i.e.,

TABLE 6-1

AFTI/F-16 DATA AT FLIGHT CONDITION (MACH=0.8, ALTITUDE=10000. FEET)

$V_T = 861.9$ (FT/SEC)	$T_s = 20.0$ (1/SEC)
$\alpha_0 = 1.66$ (DEGREES)	$b = 30.0$ (FT)
$U_0 = V_t \cdot \cos\alpha_0$ (FT/SEC)	$L_w = 1750.$ (FT)
$W_0 = V_t \cdot \cos\alpha_0$ (FT/SEC)	$\sigma_w = 5.0$ (FT/SEC)
$g = 32.174$ (FT/SEC^2)	

$Z_\alpha = -1.70$ (1/SEC)	$M_\alpha = 5.93$ (1/SEC)	$X_\alpha = 36.7$ (FT/SEC^2)
$Z_{\dot\alpha} = 2.73E-3$ (-)	$M_{\dot\alpha} = -0.303$ (1/SEC)	$X_{\dot\alpha} = -0.0683$ (FT/SEC)
$Z_q = -6.45E-3$ (-)	$M_q = -0.668$ (1/SEC)	$X_q = 0.161$ (FT/SEC)
$Z_u = -8.57E-5$ (1/FT)	$M_u = -1.34E-3$ (1/SEC-FT)	$X_u = -0.0143$ (1/SEC)
$Z_{\delta_{HT}} = -0.179$ ((1/SEC)/RAD)	$M_{\delta_{HT}} = -25.3$ ((1/SEC^2)/RAD)	$X_{\delta_{HT}} = 0.134$ ((FT/SEC^2)/RAD)
$Z_{\delta_{TEF}} = -0.295$ ((1/SEC)/RAD)	$M_{\delta_{TEF}} = -5.88$ ((1/SEC^2)/RAD)	$X_{\delta_{TEF}} = 1.46$ ((FT/SEC^2)/RAD)

in practice, both σ_w and L_w are computed by a subroutine and their values are independent of the specific aircraft modeled.

The form of the state equation is

$$\underset{\sim}{\dot{\underline{x}}}_t = \underline{A}_t \underset{\sim}{\underline{x}}_t + \underline{B}_t \underline{u}_t + \underline{G}_t \underset{\sim}{\underline{w}}_t \tag{6-31}$$

where the input vector is

$$\underline{u}_t = \begin{bmatrix} \delta_{HT_I} \\ \delta_{TEF_I} \end{bmatrix} \tag{6-32}$$

and $\underset{\sim}{\underline{w}}_t$ is actually the scalar $\underset{\sim}{\eta}_\alpha$ of equation (6-26), and of strength $Q_t=1$.

The matrices $\underline{B}_t$ and $\underline{G}_t$ are simply

$$\underline{B}_t = \left[\begin{array}{cc} 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ \hdashline T_s & 0. \\ 0. & T_s \\ \hdashline 0. & 0. \\ 0. & 0. \\ 0. & 0. \end{array}\right] \tag{6-33}$$

and

$$\underline{G}_t = \left[\begin{array}{c} 0. \\ 0. \\ 0. \\ 0. \\ \hline 0. \\ 0. \\ \hline 1. \\ a_n \\ q_n \end{array}\right] \tag{6-34}$$

Note the element "q_n" of $\underline{G}_t$ is due to the $\dot{\alpha}_g$ term in the pitch gust equation (equation (6-27c)) and has the value

$$q_n = a_{g_d} \cdot a_n \tag{6-35}$$

where a_{g_d} and a_n are defined by equations (6-28e) and (6-28d), respectively.

The measurement equation is of the form

$$\underset{\sim}{z}_t(t_i) = \underline{H}_t \underset{\sim}{x}_t(t_i) + \underset{\sim}{v}_t(t_i) \tag{6-36}$$

where it is assumed that states θ, α, and q are measured and the angle-of-attack measurement includes effects of the gust state α_g, i.e., measured angle-of-attack is $\alpha_m = \alpha + \alpha_g$+noise (modeled as white). Values for the measurement noises for the AFTI/F-16 sensors were not available. Sensor noises reported in Reference 16 were used instead (as suggested by the author of that paper, the values determined from ground and flight test for each sensor were averaged) to establish $\underline{R}_t$, the covariance of $\underset{\sim}{v}_t$ in equation (6-36) above.

The values of the matrices defining the truth model AFTI(S4,A2,G3) are (from equations (6-29) through (6-36), the data of Table 6-1, and the gust model employing Level 1 characteristics)

$$\underline{A}_t = \left[\begin{array}{cccc|cc|ccc}
0. & 0. & 1. & 0. & 0. & 0. & 0. & 0. & 0. \\
-1.085E-3 & -1.705 & 0.9963 & -8.594E-5 & -0.1795 & -0.2958 & 0. & -1.705 & -6.468E-3 \\
3.287E-4 & 6.447 & -0.9699 & -1.314E-3 & -25.25 & -5.790 & 0. & 6.447 & -0.6660 \\
-32.16 & 36.82 & -24.87 & -1.429E-2 & 0.1463 & 1.480 & 0. & 36.82 & 0.1614 \\
\hline
0. & 0. & 0. & 0. & -20.00 & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & -20.00 & 0. & 0. & 0. \\
\hline
0. & 0. & 0. & 0. & 0. & 0. & -0.4925 & 0. & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & -1.468E-3 & -0.4925 & 0. \\
0. & 0. & 0. & 0. & 0. & 0. & -3.312E-2 & -11.11 & -22.56
\end{array}\right] \tag{6-37a}$$

$$\underline{B}_t = \left[\begin{array}{cc} 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ \hline 20.00 & 0. \\ 0. & 20.00 \\ \hline 0. & 0. \\ 0. & 0. \\ 0. & 0. \end{array}\right] \tag{6-37b}$$

$$\underline{G}_t = \left[\begin{array}{c} 0. \\ 0. \\ 0. \\ 0. \\ \hline 0. \\ 0. \\ \hline 1.000 \\ 7.052\text{E-}3 \\ 0.1591 \end{array}\right] \tag{6-37c}$$

$$\underline{Q}_t = 1.000 \tag{6-37d}$$

$$\underline{H}_t = \left[\begin{array}{cccc|cc|ccc} 1.000 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.000 & 0. & 0. & 0. & 0. & 0. & 1.000 & 0. \\ 0. & 0. & 1.000 & 0. & 0. & 0. & 0. & 0. & 0. \end{array}\right] \tag{6-37e}$$

$$\underline{R}_t = \begin{bmatrix} 4.760E-6 & 0. & 0. \\ 0. & 1.220E-5 & 0. \\ 0. & 0. & 3.220E-5 \end{bmatrix} \qquad (6-37f)$$

and

$$\underline{T}_{DT} = \left[\begin{array}{cccc|cc|ccc} 1.000 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.000 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 1.000 & 0. & 0. & 0. & 0. & 0. & 0. \\ \hline 0. & 0. & 0. & 0. & 1.000 & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 1.000 & 0. & 0. & 0. \\ \hline 0. & 0. & 0. & 0. & 0. & 0. & 1.000 & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.000 & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.000 \end{array}\right] \qquad (6-37g)$$

The indicated $\underline{T}_{DT}$ matrix (see equation (3-21a)) is with respect to the design model AFTI(S3,A2,G3) described in the next subsection.

6.4.3 AFTI/F-16 Model Simplifications: Design Models. Several different levels of simplification were employed in defining reduced models of the relevant dynamics. Simplifications included deletion of a system state, elimination of control surface actuator models, and deletion of the gust states. None of these simplifications were necessary--they were effected in order to evaluate specific aspects of the design. For the various reduced models used for design (design models), other models were used as "truth models" according to the design attribute under consideration (e.g., effect of actuator lags, robustness of response when subject to design model errors). The models are discussed below in order of increasing simplification. Output equations for each model are given in the next subsection according to the controller to be designed (pitch-pointing or pitch rate). All models in which the forward velocity and gust states are absent are considered to be in angular units of degrees for convenience in presentation of time responses. Models AFTI(S4,A2,G3) and AFTI(S3,A2,G3) employ angular units of radians.

6.4.3.1 Model AFTI(S3,A2,G3). The first reduction of the truth model AFTI(S4,A2,G3) is achieved by deleting the system state equation for the perturbation in forward velocity (u) and by not incorporating stability derivatives in the angle-of-attack rate ($\dot{\alpha}$). Deletion of the forward velocity state from the design model is desirable since in the usual case for flight control designs feedback of the u state is not appropriate. The input vector and noise process scalar are as for AFTI(S4,A2,G3). The state vector is simply

$$\underline{x} = \left[\begin{array}{c} \theta \\ \alpha \\ q \\ \hline \delta_{HT} \\ \delta_{TEF} \\ \hline \alpha_g' \\ \alpha_g \\ q_g \end{array}\right] \qquad (6\text{-}38)$$

This reduction is equivalent to employing the matrix $\underline{A}_t'$ of equation (6-29) with row 4 and column 4 deleted and with equation (6-30d) computed as

$$\underline{A}_t'(8,I) = \underline{A}_t'(8,I) + a_{g_d} \cdot \underline{A}_t'(7,I) \qquad (6\text{-}39)$$

with the index "I" ranging from 1 to 8. In addition, row 4 of the matrices $\underline{B}_t$ and $\underline{G}_t$ is deleted and column 4 of the matrix $\underline{H}_t$ is deleted. As noted in the previous subsection, matrix $\underline{T}_{DT}$ of equation (6-37g) gives the correspondence of states between the truth model AFTI(S4,A2,G3) and this design model.

The parameters defining AFTI(S3,A2,G3) are computed in subroutines incorporated into the object file of CGTPIF. The values of the matrices are given below. Note that since disturbances are included in the system state, matrices $\underline{E}_x$, $\underline{A}_n$, $\underline{G}_n$, $\underline{Q}_n$, $\underline{E}_y$, and $\underline{H}_n$ of the design model do not exist. Also, the matrix $\underline{T}_{NT}$ of the truth model (equation (3-21b)) does not exist, since the equivalences for the disturbance states are included in $\underline{T}_{DT}$ (equation 6-37g)).

$$\underline{A} = \left[\begin{array}{ccc:cc}
0. & 0. & 1.000 & 0. & 0. \\
-1.082\mathrm{E}{-3} & -1.700 & 0.9936 & -0.1790 & -0.2950 \\
0. & 5.930 & -0.6680 & -25.30 & -5.880 \\
\hdashline
0. & 0. & 0. & -20.00 & 0. \\
0. & 0. & 0. & 0. & -20.00 \\
\hdashline
0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0.
\end{array}\right.$$

$$\left.\begin{array}{:ccc}
0. & 0. & 0. \\
0. & -1.700 & -6.450\mathrm{E}{-3} \\
0. & 5.930 & -0.6680 \\
\hdashline
0. & 0. & 0. \\
0. & 0. & 0. \\
\hdashline
-0.4925 & 0. & 0. \\
-1.468\mathrm{E}{-3} & -0.4925 & 0. \\
-3.312\mathrm{E}{-2} & -11.11 & -22.56
\end{array}\right] \tag{6-40a}$$

$$\underline{B} = \left[\begin{array}{cc}
0. & 0. \\
0. & 0. \\
0. & 0. \\
\hdashline
20.00 & 0. \\
0. & 20.00 \\
\hdashline
0. & 0. \\
0. & 0. \\
0. & 0.
\end{array}\right] \tag{6-40b}$$

$$G = \left[\begin{array}{l} 0. \\ 0. \\ 0. \\ \hdashline 0. \\ 0. \\ \hdashline 1.000 \\ 7.052E-3 \\ 0.1591 \end{array}\right] \tag{6-40c}$$

$$Q = 1. \tag{6-40d}$$

$$\underline{H} = \left[\begin{array}{lll:ll:lll} 1.000 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.000 & 0. & 0. & 0. & 0. & 1.000 & 0. \\ 0. & 0. & 1.000 & 0. & 0. & 0. & 0. & 0. \end{array}\right] \tag{6-40e}$$

and

$$\underline{R} = \left[\begin{array}{lll} 4.760E-6 & 0. & 0. \\ 0. & 1.220E-5 & 0. \\ 0. & 0. & 3.220E-5 \end{array}\right] \tag{6-40f}$$

Design model AFTI(S3,A2,G3) is used for design of a pitch-pointing CGT/PI controller and the Kalman filter. For this purpose, deletion of the u state is a desirable simplification in practice, since this ensures that it will not be used as a feedback variable. However, elimination of coefficients in $\dot{\alpha}$ is not actually necessary but was done so that the design and truth models differed both in dimension (u state deletion) and in specific matrix elements.

6.4.3.2 Model AFTI(S3,A2). The model AFTI(S3,A2) is used only in CGT/PI design and assumes full state feedback is available. Since a Kalman filter is not designed for this case the matrices of the measurement equation (equation (3-6)) are not needed. Also, the gust states and associated driving noise are deleted from the system state equation to facilitate consideration strictly of aircraft and actuator dynamics. Since the model entails few matrices, it is convenient to enter it directly from the terminal. The state vector is

$$\underline{x} = \left[\begin{array}{c} \theta \\ \alpha \\ q \\ \hdashline \delta_{HT} \\ \delta_{TEF} \end{array}\right] \tag{6-41}$$

and the input vector is the same as for AFTI(S4,A2,G3) and AFTI(S3,A2,G3).

The values of the matrices are

$$A = \left[\begin{array}{ccc:cc} 0. & 0. & 1.00 & 0. & 0. \\ -1.08E-3 & -1.70 & 0.994 & -0.179 & -0.295 \\ 0. & 5.93 & -0.668 & -25.3 & -5.88 \\ \hdashline 0. & 0. & 0. & -20.0 & 0. \\ 0. & 0. & 0. & 0. & -20.00 \end{array}\right] \tag{6-42a}$$

and

$$\underline{B} = \left[\begin{array}{cc} 0. & 0. \\ 0. & 0. \\ 0. & 0. \\ \hline 20.0 & 0. \\ 0. & 20.0 \end{array}\right] \tag{6-42b}$$

6.4.3.3 Model AFTI(S3). The simplest model used was AFTI(S3), which includes only the three system states of the simplified models discussed above. This reduction provides the capability to evaluate the design for the simplest aircraft representation and judge the effects of actuator lags. The state vector is

$$\underline{x} = \begin{bmatrix} \theta \\ \alpha \\ q \end{bmatrix} \tag{6-43a}$$

and the input vector is

$$\underline{u} = \begin{bmatrix} \delta_{HT} \\ \delta_{TEF} \end{bmatrix} \tag{6-43b}$$

The matrices are conveniently entered directly at the computer terminal and are

$$\underline{A} = \begin{bmatrix} 0. & 0. & 1.00 \\ 1.08\text{E}-3 & -1.70 & 0.994 \\ 0. & 5.93 & -0.668 \end{bmatrix} \tag{6-44a}$$

and

$$\underline{B} = \begin{bmatrix} 0. & 0. \\ -0.179 & -0.295 \\ -25.3 & -5.88 \end{bmatrix} \qquad (6\text{-}44b)$$

Design model AFTI(S3) is used for the CGT/PI controller designs. Effects of actuator errors in modeling (or parameter variations in flight) are evaluated by defining three different "truth models" of the same order as AFTI(S3) but with different matrix coefficients. The design based upon AFTI(S3) is not modified in performing the evaluations with respect to the various "truth models." Two of the models were obtained by modifying several of the stability derivatives at the given flight condition, while the third represents a markedly different flight condition. These will now be delineated in detail.

6.4.3.3.1 Truth Model $TM(S3)^+$. Truth model $TM(S3)^+$ was obtained from model AFTI(S3) by increasing the values of the derivatives Z_α, M_α, and M_q by about 20%. The resulting dynamics matrix is

$$\underline{A}_t = \begin{bmatrix} 0. & 0. & 1.00 \\ -1.08\text{E}-3 & -2.10 & 0.994 \\ 0. & 7.20 & -0.850 \end{bmatrix} \qquad (6\text{-}45)$$

The control matrix $\underline{B}_t$ is identical to $\underline{B}$ of equation (6-43b).

6.4.3.3.2 Truth Model TM(S3)$^-$. Truth model TM(S3)$^-$ was obtained from model AFTI(S3) by decreasing the values of the derivatives Z_α, M_α, and M_q by about 20%. The resulting dynamics matrix is

$$\underline{A}_t = \begin{bmatrix} 0. & 0. & 1.00 \\ -1.08E-3 & -1.30 & 0.994 \\ 0. & 4.80 & -0.550 \end{bmatrix} \qquad (6\text{-}46)$$

The control matrix $\underline{B}_t$ is identical to $\underline{B}$ of equation (6-43b).

6.4.3.3.3 Truth Model TM(S3)$'$. Truth model TM(S3)$'$ was obtained using the same simplifications as in AFTI(S3) but employing data for a flight condition of Mach 0.6 at an altitude of 30000. feet. Substantial changes in both the stability and control derivatives result from such an extreme change in nominal flight condition. Useful implementations would employ gain scheduling in order to accommodate such large changes in aircraft operating point. The dynamics and control matrices become

$$\underline{A}_t = \begin{bmatrix} 0. & 0. & 1.00 \\ -4.43E-3 & -0.666 & 0.997 \\ 0. & 0.500 & -0.274 \end{bmatrix} \qquad (6\text{-}47a)$$

$$\underline{B}_t = \begin{bmatrix} 0. & 0. \\ -6.06E-2 & -0.112 \\ -5.82 & -0.219 \end{bmatrix} \qquad (6\text{-}47b)$$

6.4.4 Model Output Equations

6.4.4.1 Output Equations for Pitch-Pointing Control. Pitch-pointing control entails output response in pitch attitude (θ) while constraining flight-path angle at a constant value. Referring to the flight-path angle as γ and employing the usual approximation derived for wings-level flight (Ref 30) that

$$\gamma \approx \theta - \alpha \qquad (6\text{-}48)$$

yields the following output matrices (see equation (3-5)): For AFTI(S3,A2,G3),

$$\underline{C} = \left[\begin{array}{ccc|cc|ccc} 1.000 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 1.000 & -1.000 & 0. & 0. & 0. & 0. & 0. & 0. \end{array}\right] \qquad (6\text{-}49a)$$

For AFTI(S3,A2),

$$\underline{C} = \left[\begin{array}{ccc|cc} 1.000 & 0. & 0. & 0. & 0. \\ 1.000 & -1.000 & 0. & 0. & 0. \end{array}\right] \qquad (6\text{-}49b)$$

and for AFTI(S3),

$$\underline{C} = \begin{bmatrix} 1.000 & 0. & 0. \\ 1.000 & -1.000 & 0. \end{bmatrix} \qquad (6\text{-}49c)$$

The matrices $\underline{D}_y$ and $\underline{E}_y$ of equation (3-5) are both zero matrices for all models.

6.4.4.2 Output Equations for Pitch Rate Control. Pitch rate control simply entails output response in pitch rate with no constraints on other potential output variables. Thus the output matrix actually specifies only one output (q), while the second output row is included only to maintain dimensional compatibility with the control input dimension:

For AFTI(S3,A2,G3),

$$\underline{C} = \left[\begin{array}{ccc|cc|ccc} 0. & 0. & 1.000 & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \end{array}\right] \tag{6-50a}$$

For AFTI(S3,A2),

$$\underline{C} = \left[\begin{array}{ccc|cc} 0. & 0. & 1.000 & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. \end{array}\right] \tag{6-50b}$$

and for AFTI(S3),

$$\underline{C} = \left[\begin{array}{ccc} 0. & 0. & 1.000 \\ 0. & 0. & 0. \end{array}\right] \tag{6-50c}$$

The matrices $\underline{D}_y$ and $\underline{E}_y$ of equation (3-5) are both zero matrices for all models.

6.4.5 Command Models (Equations (3-24) and (3-25))

6.4.5.1 Command Model for Pitch-Pointing Control. Pitch-pointing control is readily specified as requiring decoupled output response with respect to each of the individual command input channels. Such response characteristics are conveniently formulated entirely in terms of diagonal command model matrices to achieve first-order response trajectories. The general form in this case (of two inputs and outputs) is,

$$\underline{A}_m = \begin{bmatrix} a_1 & 0. \\ 0. & a_2 \end{bmatrix} \tag{6-51a}$$

$$\underline{B}_m = \begin{bmatrix} b_1 & 0. \\ 0. & b_2 \end{bmatrix} \tag{6-51b}$$

and

$$\underline{C}_m = \begin{bmatrix} c_1 & 0. \\ 0. & c_2 \end{bmatrix} \tag{6-51c}$$

The matrix $\underline{D}_m$ is the zero matrix.

For this command model the transfer functions for outputs $y_m(1)$ and $y_m(2)$ are:

For input $u_m(1)$,

$$\frac{y_m(1)}{u_m(1)} = \frac{c_1 \cdot b_1}{s - a_1} \tag{6-52a}$$

$$\frac{y_m(2)}{u_m(1)} = 0. \tag{6-52b}$$

and for input $u_m(2)$,

$$\frac{y_m(1)}{u_m(2)} = 0. \tag{6-53a}$$

$$\frac{y_m(2)}{u_m(2)} = \frac{c_2 \cdot b_2}{s - a_2} \qquad (6\text{-}53b)$$

The values of a_1, a_2, b_1, b_2, c_1, and c_2 may all be chosen according to the specific responses desired. Of course, a_1 and a_2 must both be chosen as negative for stable output response. The products $(c_1 \cdot b_1)$ and $(c_2 \cdot b_2)$ will affect the values of the CGT feedforward gains on the command model inputs $u_m(1)$ and $u_m(2)$, respectively (equations (6-52a) and 6-53b)). The specific choices of c_1, b_1, and c_2, b_2 values impact the CGT gains and the system response only through their respective products, i.e., scaling similarity transformations do not affect results; thus their individual values are arbitrary and selection according to convenience is appropriate. Finally, decoupling is generally not perfectly achievable. But in any CGT design in which the PI gains are fixed in value, the ratio of the steady-state magnitude of the commanded output to the maximum transient magnitude achieved in the nominally zero output channel is constant for all values of c_1, b_1, c_2, and b_2 since the feedforward gains for each command input channel are independent for command models defining decoupled output response.

From the definition of the output matrix of equation (6-49), output 1 is the desired pitch-pointing output. The desired form of response is thus defined by equations (6-52a,b); command input $u_m(1)$ then commands pitch-pointing.

The values of a_1, b_1, and c_1 must be selected to give desirable response while the values of a_2, b_2, and c_2 are arbitrary.

The value of a_1 was selected as -5., giving a first-order pitch-pointing command model response with a rise-time of about 0.6 seconds for a step command input. When used in conjunction with design models AFTI(S3) and AFTI(S3,A2) the numerator of equation (6-52a) was selected to give an output to input ratio of unity at low frequency ($c_1 \cdot b_1 = |a_1|$). However, since the angular units for design model AFTI(S3,A2,G3) are radians to correspond to truth model AFTI(S4,A2,G3), by design choice a command model output to input ratio of 0.02 was selected for designs with AFTI(S3,A2,G3) ($c_1 \cdot b_1 = 0.02 \cdot |a_1|$). The values of a_2, b_2, and c_2 were arbitrarily set equal to the values of a_1, b_1, and c_1, respectively.

For all pitch-pointing designs the command model's dynamics matrix is

$$\underline{A}_m = \begin{bmatrix} -5. & 0. \\ 0. & -5. \end{bmatrix} \tag{6-54}$$

In conjunction with design models AFTI(S3) and AFTI(S3,A2), the matrices $\underline{B}_m$ and $\underline{C}_m$ are

$$\underline{B}_m = \begin{bmatrix} 5. & 0. \\ 0. & 5. \end{bmatrix} \tag{6-55a}$$

and

$$\underset{\sim}{C}_m = \begin{bmatrix} 1. & 0. \\ 0. & 1. \end{bmatrix} \qquad (6\text{-}55b)$$

In conjunction with design model AFTI(S3,A2,G3), the matrices $\underset{\sim}{B}_m$ and $\underset{\sim}{C}_m$ are

$$\underset{\sim}{B}_m = \begin{bmatrix} 0.1 & 0. \\ 0. & 0.1 \end{bmatrix} \qquad (6\text{-}56a)$$

and

$$\underset{\sim}{C}_m = \begin{bmatrix} 1. & 0. \\ 0. & 1. \end{bmatrix} \qquad (6\text{-}56b)$$

Matrix $\underset{\sim}{D}_m$ is the zero matrix.

6.4.5.2 Command Model for Pitch Rate Control. The same command model form described above was also used for the pitch rate controller designs. In this case the product of c_2 and b_2 was selected as zero since only the first of the two outputs is actually specified in the design model output equation. The pitch rate controller was designed using only the AFTI(S3) and AFTI(S3,A2) design models and an output to input response ratio of unity was chosen. The matrices defining the pitch rate command model are

$$\underset{\sim}{A}_m = \begin{bmatrix} -5. & 0. \\ 0. & -5. \end{bmatrix} \qquad (6\text{-}57a)$$

$$\underline{B}_m = \begin{bmatrix} 5. & 0. \\ 0. & 5. \end{bmatrix} \qquad (6\text{-}57b)$$

and

$$\underline{C}_m = \begin{bmatrix} 1. & 0. \\ 0. & 0. \end{bmatrix} . \qquad (6\text{-}57c)$$

Matrix $\underline{D}_m$ is the zero matrix.

6.4.6 <u>CGT/PI Pitch-Pointing Controller Design</u>. CGT/PI controller designs achieving pitch-pointing control are described briefly below. Designs are determined for design models AFTI(S3), AFTI(S3,A2), and AFTI(S3,A2,G3). An important observation is that designs which do not account for the lags introduced by the control surface actuators are unstable when required to control the aircraft through actuators with the lags of the AFTI/F-16 aircraft. Designs which did include actuator models were stable but did show response behavior somewhat inferior to that achievable without actuators. When the actuators are modeled as having lag time-constants several times smaller than those of AFTI/F-16 (τ=0.05 seconds), designs which neglected actuator dynamics behaved very well when controlling the aircraft through the actuators.

6.4.6.1 <u>AFTI(S3) Pitch-Pointing CGT/PI</u>. Consistent with usual experience in aircraft flight control designs for pitch motion, the utility of pitch rate damping was quickly determined in the PI and CGT/PI designs for the pitch-pointing mode. Figure 6-10 illustrates the response of an initial PI regulator design with quadratic weights of 200. on both outputs and 1. on input magnitudes and rates. The initial conditions were 1. on output 1 and 2. on output 2. Plot symbols 1 and 2 are outputs 1 and 2, respectively. The response in pitch attitude (output 1) is clearly poorly damped. The CGT/PI controller response with the same PI regulator is shown in Figure 6-11. Plot symbols 1 and 2 are the aircraft's actual and desired pitch attitude response and symbols 3 and 4 are the aircraft's actual and desired flight-path angle deviations. Compared to the desired response, the actual pitch-pointing response initially lags, then overshoots, and finally oscillates about the final value. The alternate channel response is small but oscillatory. While the response is good overall, clearly improved pitch damping is needed.

Figures 6-12 and 6-13 show the corresponding responses for a new PI regulator design. This final PI design employs identical weightings to that discussed above, but also includes a weight of 50. on the pitch rate state. Note the dramatic improvement in the PI regulator's response to initial conditions both in speed of response and damping of oscillations. The pitch-pointing response of the CGT/PI controller shown in Figure 6-13 is nearly an exact match with the desired response. Also, the alternate output channel response is very small (its maximum magnitude is about 0.025 as in the previous case) and is

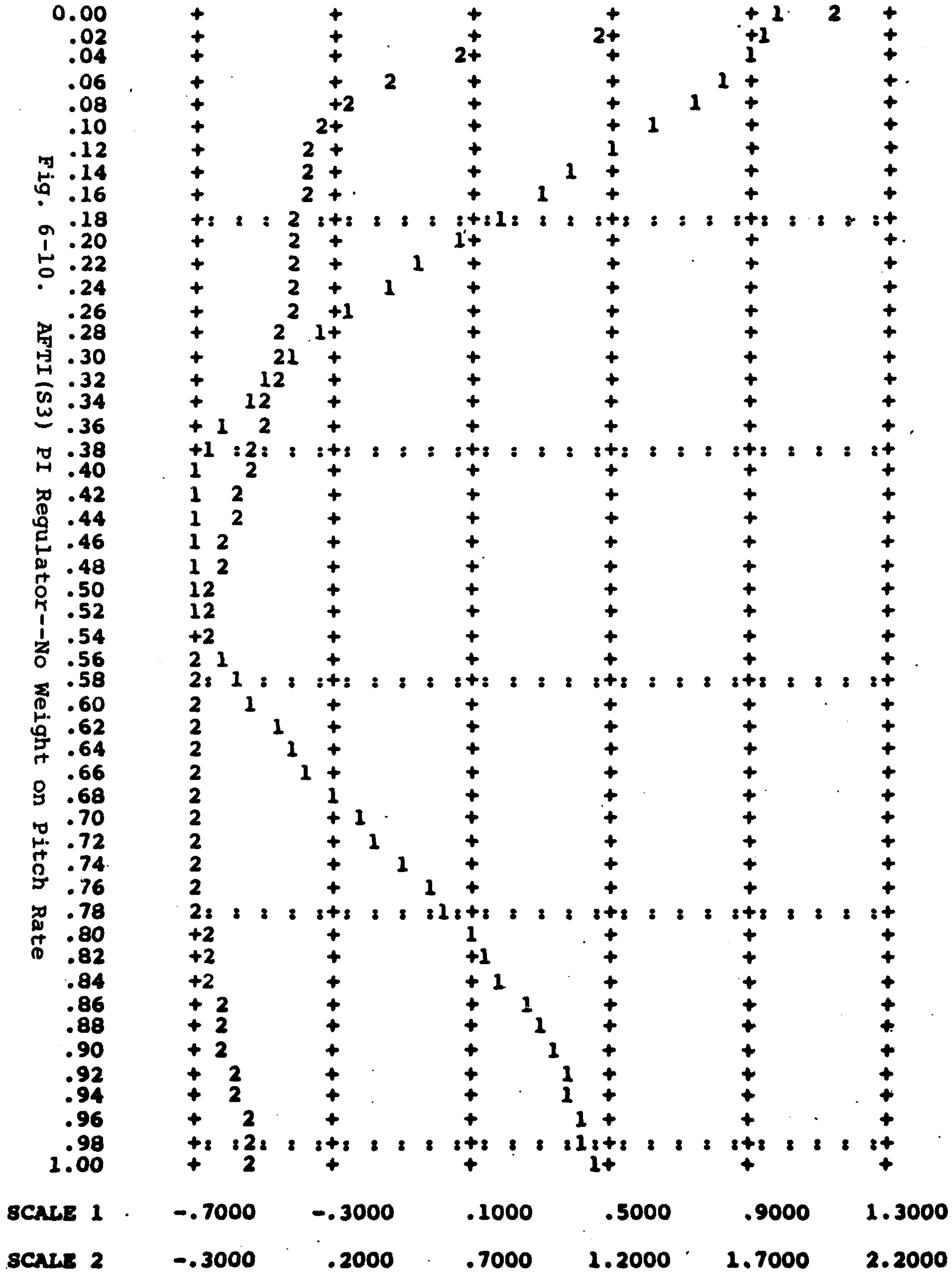

Fig. 6-10. AFTI(S3) PI Regulator--No Weight on Pitch Rate

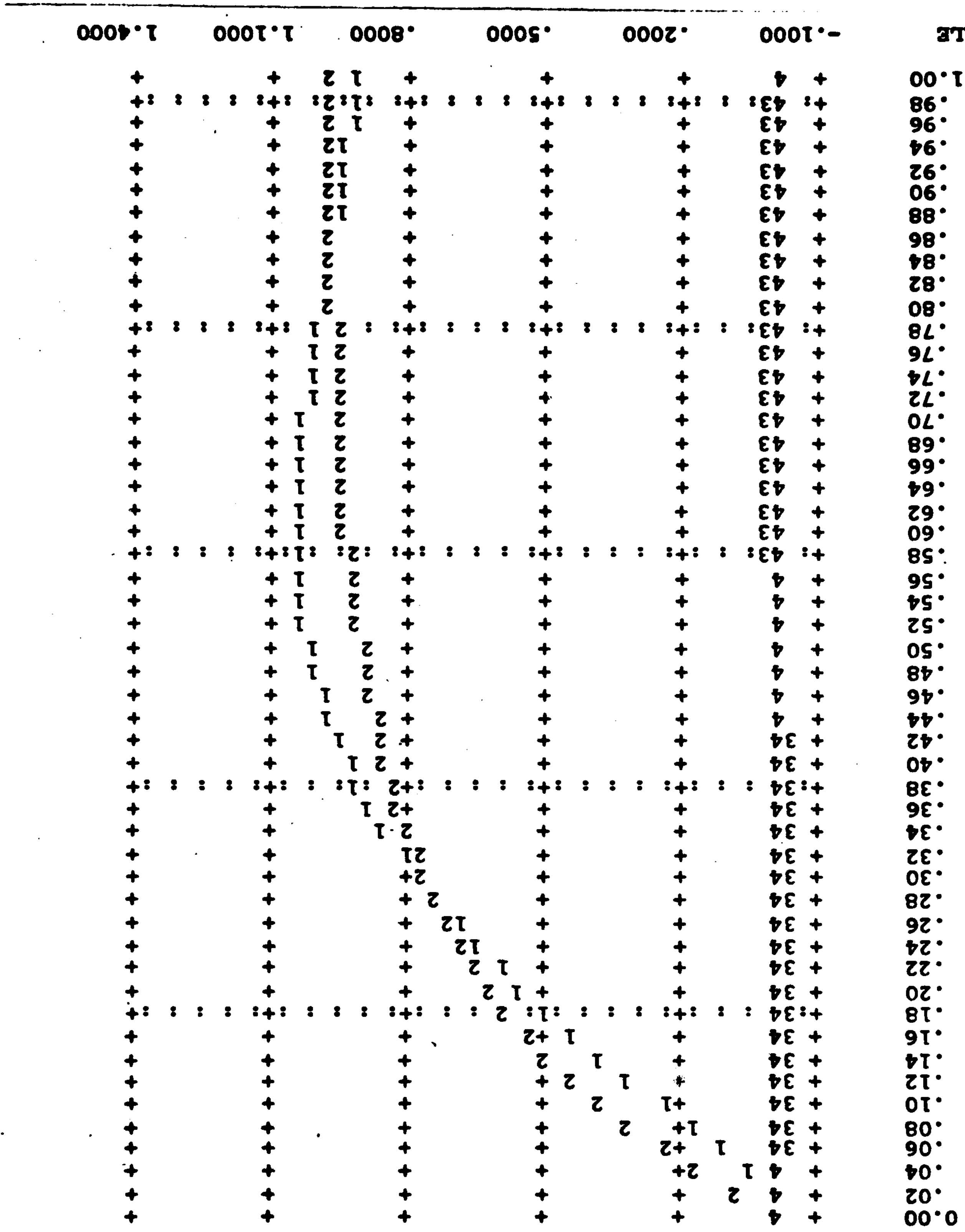

Fig. 6-11. AFTI(S3) Pitch-Pointing CGT/PI--No Weight on Pitch Rate

```
0.00      +         +         +         +         +  12     +
 .02      +         +         +        2+        1+         +
 .04      +         +        2+         1         +         +
 .06      +         +  2      1         +         +         +
 .08      +        2+ 1       +         +         +         +
 .10      +     2   +         +         +         +         +
 .12      +  12     +         +         +         +         +
 .14      + 12      +         +         +         +         +
 .16      + 21      +         +         +         +         +
 .18      +2 1 : : :+: : : : :+: : : : :+: : : : :+: : : : :+
 .20      +2  1     +         +         +         +         +
 .22      +2   1    +         +         +         +         +
 .24      +2    1   +         +         +         +         +
 .26      +2    1   +         +         +         +         +
 .28      +2     1  +         +         +         +         +
 .30      +2     1  +         +         +         +         +
 .32      +2     1  +         +         +         +         +
 .34      +2     1  +         +         +         +         +
 .36      +2     1  +         +         +         +         +
 .38      +2 : : 1 :+: : : : :+: : : : :+: : : : :+: : : : :+
 .40      +2     1  +         +         +         +         +
 .42      +2     1  +         +         +         +         +
 .44      + 2    1  +         +         +         +         +
 .46      + 2     1 +         +         +         +         +
 .48      + 2     1 +         +         +         +         +
 .50      + 2     1 +         +         +         +         +
 .52      + 2     1 +         +         +         +         +
 .54      + 2     1 +         +         +         +         +
 .56      + 2     1 +         +         +         +         +
 .58      +:2: : :1:+: : : : :+: : : : :+: : : : :+: : : : :+
 .60      + 2     1 +         +         +         +         +
 .52      + 2     1 +         +         +         +         +
 .64      + 2     1 +         +         +         +         +
 .66      + 2     1 +         +         +         +         +
 .68      + 2     1 +         +         +         +         +
 .70      + 2      1+         +         +         +         +
 .72      + 2      1+         +         +         +         +
 .74      + 2      1+         +         +         +         +
 .76      + 2      1+         +         +         +         +
 .78      +:2: : : 1+: : : : :+: : : : :+: : : : :+: : : : :+
 .80      + 2      1+         +         +         +         +
 .82      + 2      1+         +         +         +         +
 .84      + 2      1+         +         +         +         +
 .86      + 2      1+         +         +         +         +
 .88      + 2      1+         +         +         +         +
 .90      + 2      1+         +         +         +         +
 .92      + 2      1+         +         +         +         +
 .94      + 2      1+         +         +         +         +
 .96      + 2      1+         +         +         +         +
 .98      +: 2 : : 1+: : : : :+: : : : :+: : : : :+: : : : :+
1.00      +  2     1+         +         +         +         +

SCALE 1  -.3000    -.0000     .3000     .6000     .9000   1.2000

SCALE 2  -.2000     .3000     .8000   1.3000    1.8000    2.3000
```

Fig. 6-12. AFTI(S3) PI Regulator--Weight on Pitch Rate

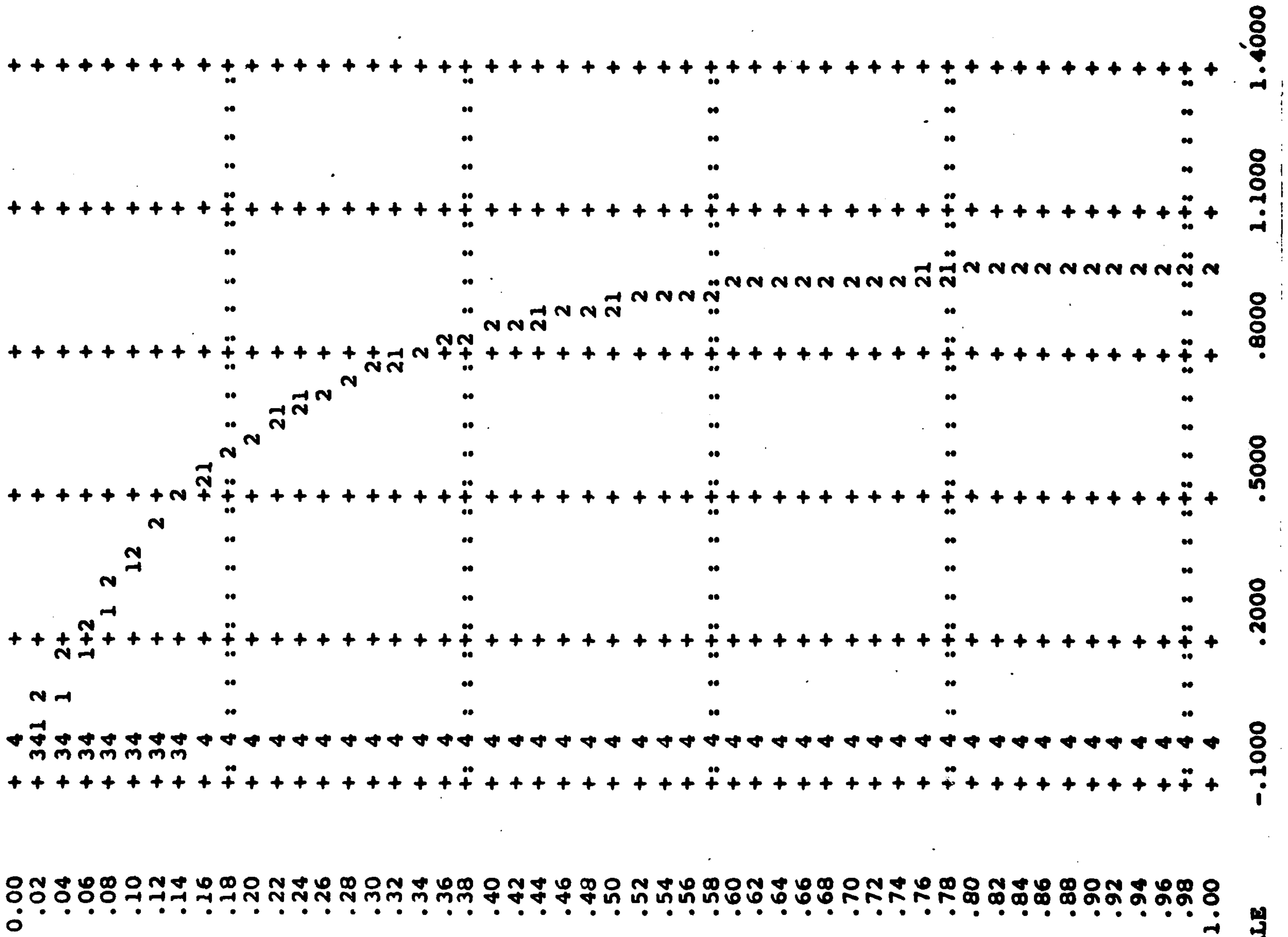

Fig. 6-13. AFTI(S3) Pitch-Pointing CGT/PI--Weight on Pitch Rate

rapidly damped (about 0.14 seconds compared to more than 1.0 seconds in the previous case). Figure 6-14 shows the responses of the control surfaces for the CGT/PI response of Figure 6-13. Plot symbols 1 and 2 are the horizontal tail and trailing-edge flaps, respectively. For a steady-state pitch-pointing command of 1.° steady-state surface deflections of about 1.7° for the tail and -6.9° for the flap are required. Note that the tail surface has an initially large deflection at the first controller sample time. As mentioned earlier in Section 6.1.2, this initial commanded deflection of large magnitude is typically the most important design constraint for the CGT/PI controller and is due to the feedforward gain $\underline{K}_{x_u}$ on the command model input. Reduction of this initial input command may require that PI gains be reduced and/or that command model dynamics be changed.

The set of quadratic weights of this PI regulator was used for all the pitch-pointing designs. The PI gains for design model AFTI(S3) are

$$\underline{K}_x = \begin{bmatrix} -41.72 & 19.47 & -1.361 \\ 76.73 & -83.11 & 0.4885 \end{bmatrix} \qquad (6\text{-}58a)$$

and

$$\underline{K}_z = \begin{bmatrix} -0.1603 & -0.8697 \\ -2.963 & 3.069 \end{bmatrix} \qquad (6\text{-}58b)$$

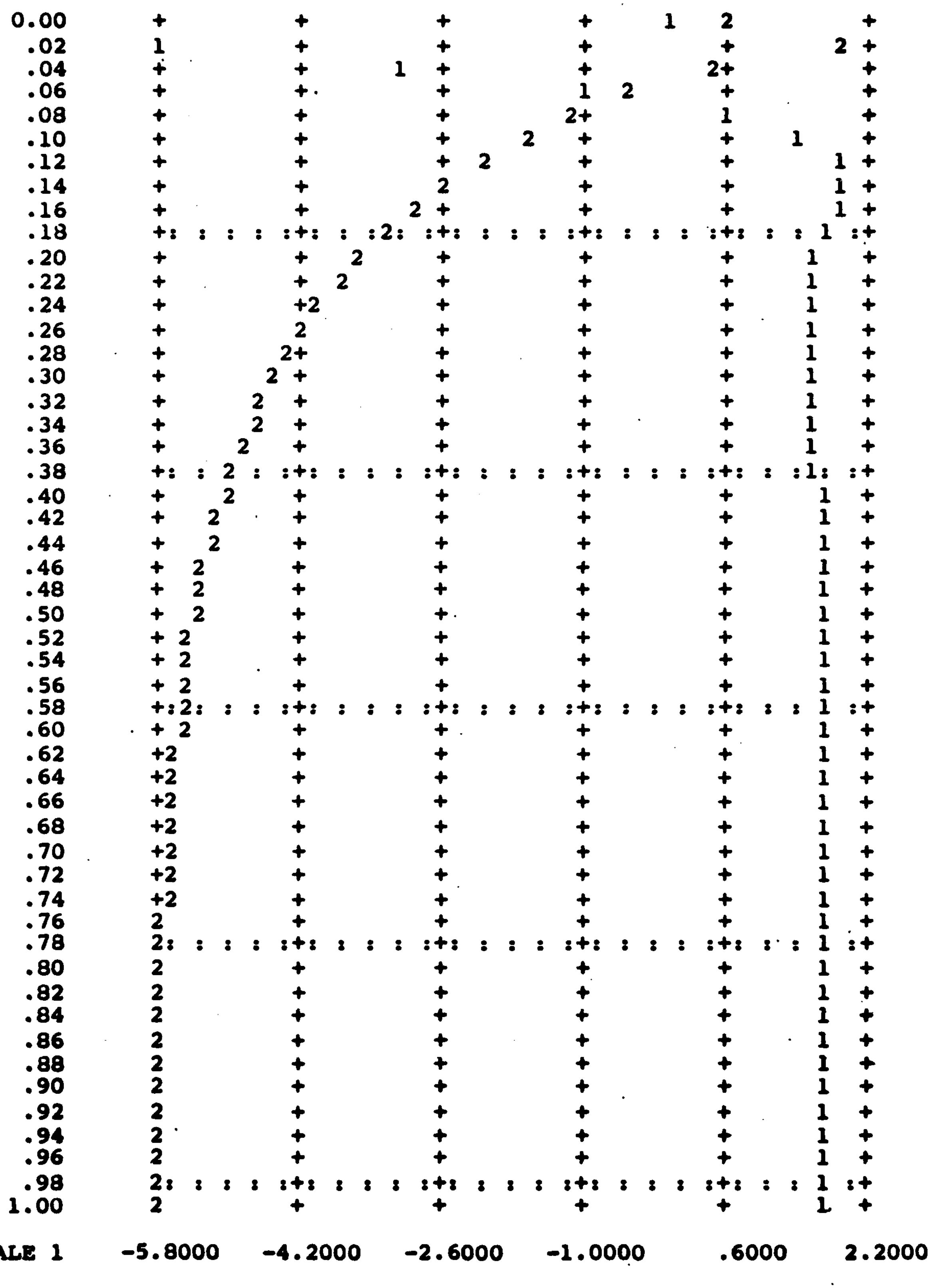

Fig. 6-14. AFTI(S3) CGT/PI Control Surface Deflections

and the feedforward CGT/PI gains are

$$\underline{K}_{x_m} = \begin{bmatrix} -14.68 & -16.85 \\ -14.67 & 70.88 \end{bmatrix} \qquad (6\text{-}59a)$$

and

$$\underline{K}_{x_u} \begin{bmatrix} -5.736 & -4.451 \\ 1.419 & 19.11 \end{bmatrix} \qquad (6\text{-}59b)$$

Note that for a unit step input on command model input 1 (pitch-pointing command input) element (1,1) of matrix $\underline{K}_{x_u}$ is the value of the command input during the first sample period following the command. Thus in order to modify this initial magnitude, designs should be sought which give different values for $\underline{K}_{x_u}$ (1,1).

The next three figures demonstrate the response of this same CGT/PI controller when the design model includes parameter errors. Figure 6-15, 6-16, and 6-17 are the CGT/PI responses evaluated using truth models $TM(S3)^{+}$, $TM(S3)^{-}$, and $TM(S3)'$, respectively. As before, plot symbols 1 and 2 are actual and desired pitch-pointing response and symbols 3 and 4 are actual and desired alternate channel response. The first two figures show that for significant errors (±20%) in key model parameters the controlled response is virtually identical to that for nominal parameter values. Figure 6-17 demonstrates that even for very great differences in parameter values due to a large change in flight condition the CGT/PI response,

```
   0.00      +  4      +         +         +         +         +
    .02      + 341 2   +         +         +         +         +
    .04      + 34  1  2+         +         +         +         +
    .06      + 34     1+2        +         +         +         +
    .08      + 34      + 1 2     +         +         +         +
    .10      + 34      +    12   +         +         +         +
    .12      + 34      +       2 +         +         +         +
    .14      +  4      +         2         +         +         +
    .16      +  4      +         +21       +         +         +
    .18      +: 4 : : :+: : : : :+: 2 : : :+: : : : :+: : : : :+
    .20      +  4      +         +   2     +         +         +
    .22      +  4      +         +    21   +         +         +
    .24      +  4      +         +     21  +         +         +
    .26      +  4      +         +      2  +         +         +
    .28      +  4      +         +       2 +         +         +
    .30      +  4      +         +        2+         +         +
    .32      +  4      +         +        21         +         +
    .34      +  4      +         +         2         +         +
    .36      +  4      +         +         +2        +         +
    .38      +: 4 : : :+: : : : :+: : : : :+2 : : : :+: : : : :+
    .40      +  4      +         +         + 2       +         +
    .42      +  4      +         +         + 2       +         +
    .44      +  4      +         +         + 21      +         +
    .46      +  4      +         +         +  2      +         +
    .48      +  4      +         +         +  2      +         +
    .50      +  4      +         +         +  21     +         +
    .52      +  4      +         +         +   2     +         +
    .54      +  4      +         +         +   2     +         +
    .56      +  4      +         +         +   2     +         +
    .58      +: 4 : : :+: : : : :+: : : : :+: :2: : :+: : : : :+
    .60      +  4      +         +         +    2    +         +
    .62      +  4      +         +         +    2    +         +
    .64      +  4      +         +         +    2    +         +
    .66      +  4      +         +         +    2    +         +
    .68      +  4      +         +         +    2    +         +
    .70      +  4      +         +         +    2    +         +
    .72      +  4      +         +         +    2    +         +
    .74      +  4      +         +         +    2    +         +
    .76      +  4      +         +         +    21   +         +
    .78      +: 4 : : :+: : : : :+: : : : :+: : 21: :+: : : : :+
    .80      +  4      +         +         +     2   +         +
    .82      +  4      +         +         +     2   +         +
    .84      +  4      +         +         +     2   +         +
    .86      +  4      +         +         +     2   +         +
    .88      +  4      +         +         +     2   +         +
    .90      +  4      +         +         +     2   +         +
    .92      +  4      +         +         +     2   +         +
    .94      +  4      +         +         +     2   +         +
    .96      +  4      +         +         +     2   +         +
    .98      +: 4 : : :+: : : : :+: : : : :+: : :2: :+: : : : :+
   1.00      +  4      +         +         +     2   +         +

SCALE       -.1000     .2000     .5000     .8000    1.1000    1.4000
```

Fig. 6-15. AFTI(S3) Pitch-Pointing CGT/PI WRT TM(S3)[+]

```
   0.00      +  4      +         +         +         +         +
    .02      + 341 2   +         +         +         +         +
    .04      + 34  1  2+         +         +         +         +
    .06      + 34     1+2        +         +         +         +
    .08      + 34      + 1 2     +         +         +         +
    .10      + 34      +    12   +         +         +         +
    .12      + 34      +       2 +         +         +         +
    .14      + 34      +         2         +         +         +
    .16      + 34      +         +21       +         +         +
    .18      +:34 : : :+: : : : :+: 2 : : :+: : : : :+: : : : :+
    .20      + 34      +         +   2     +         +         +
    .22      + 34      +         +    21   +         +         +
    .24      + 34      +         +     21  +         +         +
    .26      + 34      +         +      2  +         +         +
    .28      + 34      +         +       2 +         +         +
    .30      + 34      +         +        2+         +         +
    .32      + 34      +         +        21         +         +
    .34      + 34      +         +         2         +         +
    .36      + 34      +         +         +2        +         +
    .38      +:34 : : :+: : : : :+: : : : :+2 : : : :+: : : : :+
    .40      + 34      +         +         + 2       +         +
    .42      + 34      +         +         + 2       +         +
    .44      + 34      +         +         + 21      +         +
    .46      +  4      +         +         +  2      +         +
    .48      +  4      +         +         +  2      +         +
    .50      +  4      +         +         +  21     +         +
    .52      +  4      +         +         +   2     +         +
    .54      +  4      +         +         +   2     +         +
    .56      +  4      +         +         +   2     +         +
    .58      +: 4 : : :+: : : : :+: : : : :+: :2: : :+: : : : :+
    .60      +  4      +         +         +    2    +         +
    .62      +  4      +         +         +    2    +         +
    .64      +  4      +         +         +    2    +         +
    .66      +  4      +         +         +    2    +         +
    .68      +  4      +         +         +    2    +         +
    .70      +  4      +         +         +    2    +         +
    .72      +  4      +         +         +    2    +         +
    .74      +  4      +         +         +    2    +         +
    .76      +  4      +         +         +    21   +         +
    .78      +: 4 : : :+: : : : :+: : : : :+: : 21: :+: : : : :+
    .80      +  4      +         +         +     2   +         +
    .82      +  4      +         +         +     2   +         +
    .84      +  4      +         +         +     2   +         +
    .86      +  4      +         +         +     2   +         +
    .88      +  4      +         +         +     2   +         +
    .90      +  4      +         +         +     2   +         +
    .92      +  4      +         +         +     2   +         +
    .94      +  4      +         +         +     2   +         +
    .96      +  4      +         +         +     2   +         +
    .98      +: 4 : : :+: : : : :+: : : : :+: : :2: :+: : : : :+
   1.00      +  4      +         +         +     2   +         +

SCALE       -.1000     .2000     .5000     .8000    1.1000    1.4000
```

Fig. 6-16. AFTI(S3) Pitch-Pointing CGT/PI WRT TM(S3)

```
0.00    +  4      +         +         +         +         +
 .02    +  4  2   +         +         +         +         +
 .04    + 341    2+         +         +         +         +
 .06    + 34 1    +2        +         +         +         +
 .08    + 34  1   +   2     +         +         +         +
 .10    + 34     1+     2   +         +         +         +
 .12    +3 4      +1      2 +         +         +         +
 .14    +3 4      +   1     2         +         +         +
 .16    +3 4      +     1   +2        +         +         +
 .18    +3 4 : : :+: : : : 1+: 2 : : :+: : : : :+: : : : :+
 .20    +3 4      +         + 1 2     +         +         +
 .22    +3 4      +         +    2    +         +         +
 .24    +3 4      +         +     2 1 +         +         +
 .26    +3 4      +         +      2  1         +         +
 .28    +3 4      +         +       2 + 1       +         +
 .30    + 34      +         +        2+   1     +         +
 .32    + 34      +         +        2+    1    +         +
 .34    + 34      +         +         2     1   +         +
 .36    +  4      +         +         +2    1   +         +
 .38    +: 4 : : :+: : : : :+: : : : :+2 : : 1 :+: : : : :+
 .40    +  4      +         +         + 2    1  +         +
 .42    +  4      +         +         + 2   1   +         +
 .44    +  43     +         +         + 2   1   +         +
 .46    +  43     +         +         +  2 1    +         +
 .48    +  43     +         +         +  21     +         +
 .50    +  43     +         +         +  21     +         +
 .52    +  43     +         +         +  12     +         +
 .54    +  43     +         +         + 1 2     +         +
 .56    +  4      +         +         + 1 2     +         +
 .58    +: 4 : : :+: : : : :+: : : : :+1 :2: : :+: : : : :+
 .60    +  4      +         +         +1   2    +         +
 .62    +  4      +         +         +1   2    +         +
 .64    +  4      +         +         +1   2    +         +
 .66    + 34      +         +         +1   2    +         +
 .68    + 34      +         +         +1   2    +         +
 .70    + 34      +         +         + 1  2    +         +
 .72    + 34      +         +         + 1  2    +         +
 .74    + 34      +         +         +  1 2    +         +
 .76    + 34      +         +         +   12    +         +
 .78    +:34 : : :+: : : : :+: : : : :+: :12 : :+: : : : :+
 .80    + 34      +         +         +    12   +         +
 .82    + 34      +         +         +    12   +         +
 .84    + 34      +         +         +     2   +         +
 .86    + 34      +         +         +     2   +         +
 .88    + 34      +         +         +     21  +         +
 .90    + 34      +         +         +     21  +         +
 .92    + 34      +         +         +     21  +         +
 .94    + 34      +         +         +     21  +         +
 .96    + 34      +         +         +     21  +         +
 .98    +:34 : : :+: : : : :+: : : : :+: : :21 :+: : : : :+
1.00    + 34      +         +         +     21  +         +

SCALE  -.1000     .2000     .5000     .8000    1.1000    1.4000
```

Fig. 6-17. AFTI(S3) Pitch-Pointing CGT/PI WRT TM(S3)'

while seriously degraded, is not unusable. Clearly, gain scheduling is appropriate (and is typically employed) in order to accommodate such extremes in operating conditions.

6.4.6.2 AFTI(S3,A2) Pitch-Pointing CGT/PI. Figure 6-18 shows the response of the PI regulator determined using the quadratic weightings described in the subsection above. Model AFTI(S3,A2) is used both for design and evaluation in this case. The response shown is the analog of the response plotted in Figure 6-12. Note that inclusion of actuator dynamics leads to slightly slower and less well-damped pitch attitude response.

The corresponding CGT/PI response is shown in Figure 6-19, which is the analog of the response of Figure 6-13. The pitch attitude response is degraded in comparison to what was achieved when the inputs were assumed to be the control surface deflections directly. The maximum magnitude of the deviation in flight path angle remains small (about 0.05).

Although the controller's inputs to the actuators were nearly identical to those generated to drive the control surfaces in the AFTI(S3) design, the surface deflections actually attained in the initial phase of response were quite different. Figure 6-20 shows the deflections of the horizontal tail and trailing-edge flaps (plot symbols 1 and 2, respectively). These deflections may be compared

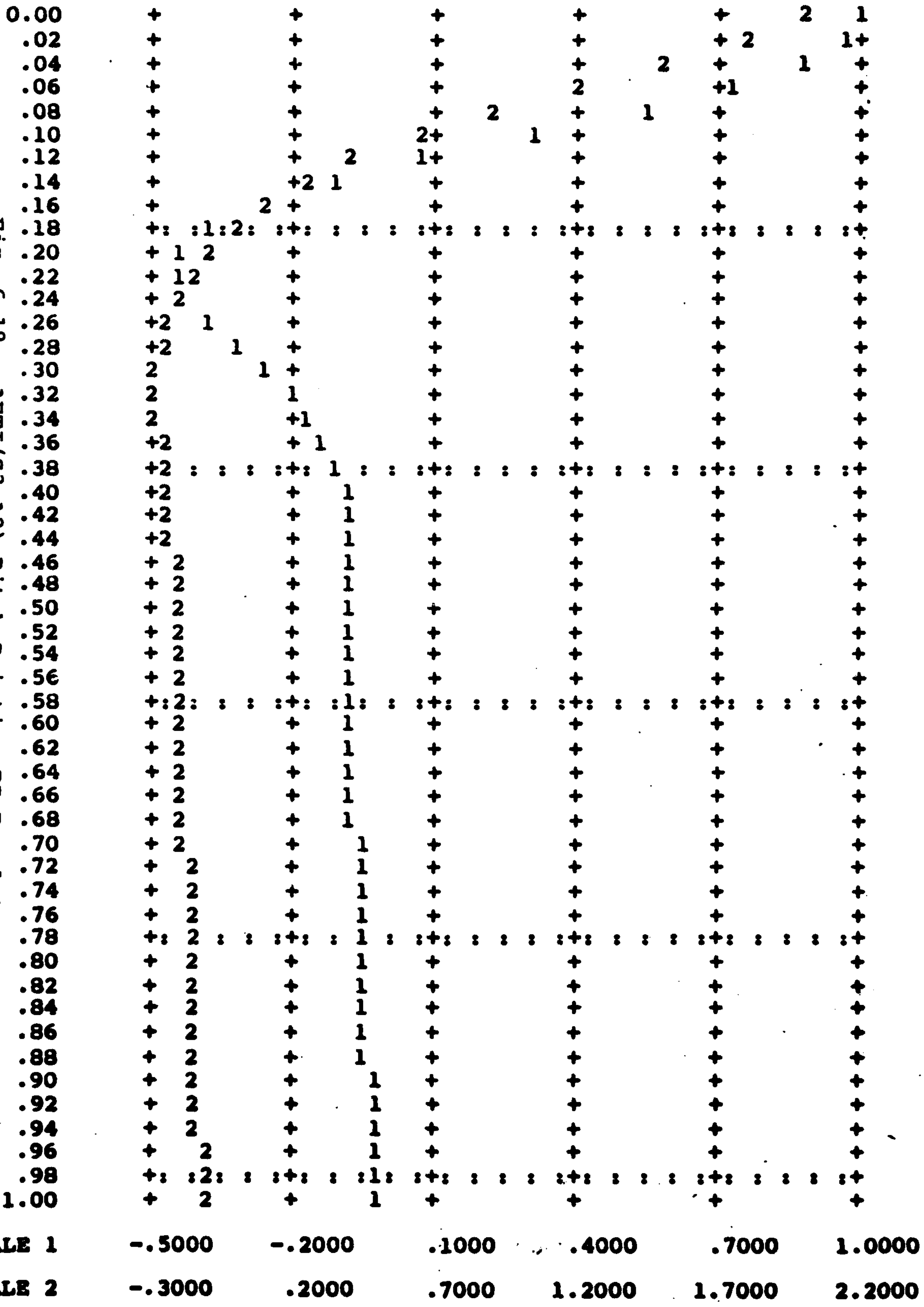

Fig. 6-18. AFTI(S3,A2) Pitch-Pointing PI Regulator

```
0.00      +  4      +         +         +         +         +
 .02      +  4  2   +         +         +         +         +
 .04      + 341    2+         +         +         +         +
 .06      + 34 1    +2        +         +         +         +
 .08      +3 4    1 +   2     +         +         +         +
 .10      +3 4      +1    2   +         +         +         +
 .12      +3 4      +   1   2 +         +         +         +
 .14      +3 4      +      1  2         +         +         +
 .16      +3 4      +         12        +         +         +
 .18      +3 4 : : :+: : : : :+: 2 : : :+: : : : :+: : : : :+
 .20      + 34      +         +   21    +         +         +
 .22      + 34      +         +    2 1  +         +         +
 .24      + 34      +         +     2  1+         +         +
 .26      + 34      +         +      2  1         +         +
 .28      +  4      +         +       2 +1        +         +
 .30      +  4      +         +        2+1        +         +
 .32      +  4      +         +        2+1        +         +
 .34      +  4      +         +         2 1       +         +
 .36      +  4      +         +         +21       +         +
 .38      +: 4 : : :+: : : : :+: : : : :+21: : : :+: : : : :+
 .40      +  4      +         +         + 2       +         +
 .42      +  4      +         +         + 21      +         +
 .44      +  4      +         +         + 21      +         +
 .46      +  4      +         +         +  2      +         +
 .48      +  4      +         +         +  2      +         +
 .50      +  4      +         +         +  21     +         +
 .52      +  4      +         +         +   2     +         +
 .54      +  4      +         +         +   2     +         +
 .56      +  4      +         +         +   21    +         +
 .58      +: 4 : : :+: : : : :+: : : : :+: :21 : :+: : : : :+
 .60      +  4      +         +         +    2    +         +
 .62      +  4      +         +         +    2    +         +
 .64      +  4      +         +         +    2    +         +
 .66      +  4      +         +         +    2    +         +
 .68      +  4      +         +         +    21   +         +
 .70      +  4      +         +         +    21   +         +
 .72      +  4      +         +         +    21   +         +
 .74      +  4      +         +         +    21   +         +
 .76      +  4      +         +         +    21   +         +
 .78      +: 4 : : :+: : : : :+: : : : :+: : 21: :+: : : : :+
 .80      +  4      +         +         +     2   +         +
 .82      +  4      +         +         +     2   +         +
 .84      +  4      +         +         +     2   +         +
 .86      +  4      +         +         +     2   +         +
 .88      +  4      +         +         +     2   +         +
 .90      +  4      +         +         +     2   +         +
 .92      +  4      +         +         +     2   +         +
 .94      +  4      +         +         +     2   +         +
 .96      +  4      +         +         +     2   +         +
 .98      +: 4 : : :+: : : : :+: : : : :+: : :2: :+: : : : :+
1.00      +  4      +         +         +     2   +         +

SCALE    -.1000     .2000     .5000     .8000   1.1000    1.4000
```

Fig. 6-19. AFTI(S3,A2) Pitch-Pointing CGT/PI

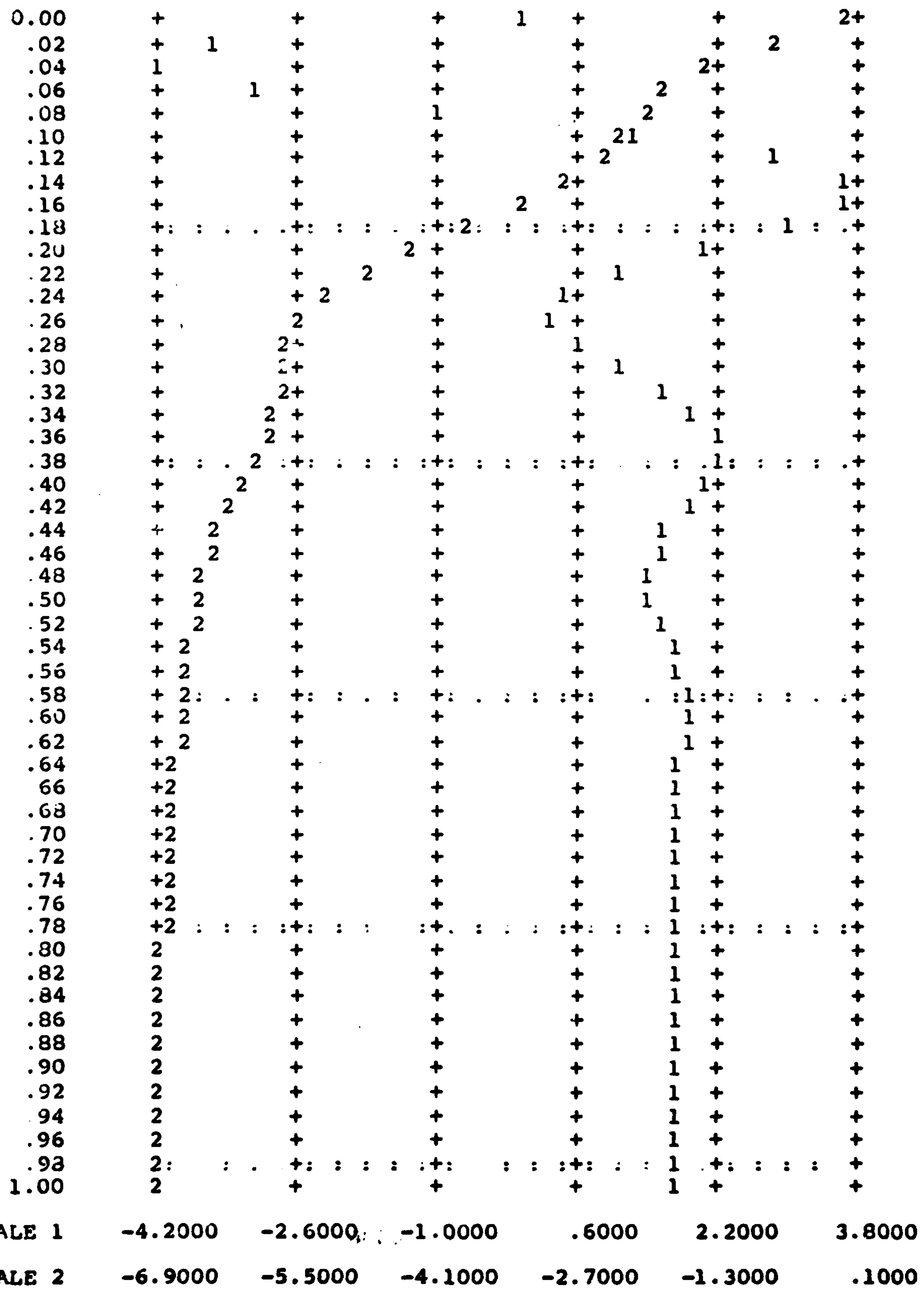

Fig. 6-20. AFTI(S3,A2) Pitch-Pointing CGT/PI Control Surface Deflections

to Figure 6-14, for which actuators were not modeled. The lags introduced by the control actuators account for the differences in the control surface deflections and in the CGT/PI controller's response during the initial transient phase of response.

The PI gains for the AFTI(S3,A2) pitch-pointing design model are

$$\underline{K}_x = \left[\begin{array}{ccc:cc} -38.87 & 19.69 & -1.687 & 1.282 & 5.712E-2 \\ 78.07 & -83.61 & 0.4227 & 5.502E-2 & 1.045 \end{array}\right] \qquad (6\text{-}60a)$$

and

$$\underline{K}_z = \begin{bmatrix} 7.00E-3 & -0.8716 \\ -2.929 & 3.068 \end{bmatrix} \qquad (6\text{-}60b)$$

and the CGT/PI feedforward gains are

$$\underline{K}_{x_m} = \begin{bmatrix} -9.487 & -14.92 \\ -18.44 & 61.13 \end{bmatrix} \qquad (6\text{-}61a)$$

and

$$\underline{K}_{x_u} = \begin{bmatrix} -5.908 & -8.554 \\ -1.059 & 36.43 \end{bmatrix} \qquad (6\text{-}61b)$$

6.4.6.3 <u>AFTI(S3,A2,G3) Pitch-Pointing CGT/PI</u>. The response of the CGT/PI controller designed based upon model AFTI(S3,A2,G3), using the same quadratic weights as previously, and evaluated with respect to the truth model AFTI(S3,A2,G3) is shown in Figure 6-21. It is the analog of the previous plots of Figure 6-13 and 6-19 for design models AFTI(S3) and AFTI(S3,A2), respectively. The response shown is in units of radians. Clearly, there is no significant degradation in response compared to that shown in Figure 6-19 for AFTI(S3,A2). Thus deletion of the forward velocity perturbation state and neglect of effects due to $\dot{\alpha}$ derivatives is of less impact than failure to model actuator dynamics.

The gains for the PI regulator are

$$\underline{K}_x = \left[\begin{array}{ccc:cc:ccc} -38.87 & 19.69 & -1.687 & 1.282 & 5.710\text{E-}2 & 0.1851 & -62.16 & 0.0294 \\ 78.03 & -83.61 & 0.4227 & 5.500\text{E-}2 & 1.045 & -0.8324 & 280.0 & 7.690\text{E-}3 \end{array}\right] \tag{6-62a}$$

and

$$\underline{K}_z = \begin{bmatrix} 7.040\text{E-}3 & -0.8717 \\ -2.929 & 3.068 \end{bmatrix} \tag{6-62b}$$

and the CGT/PI feedforward gains are

$$\underline{K}_{x_m} = \begin{bmatrix} -9.482 & -14.92 \\ -18.47 & 61.13 \end{bmatrix} \tag{6-63a}$$

and

$$\underline{K}_{x_u} = \begin{bmatrix} -0.1181 & -0.1711 \\ -0.0215 & 0.7287 \end{bmatrix} \tag{6-63b}$$

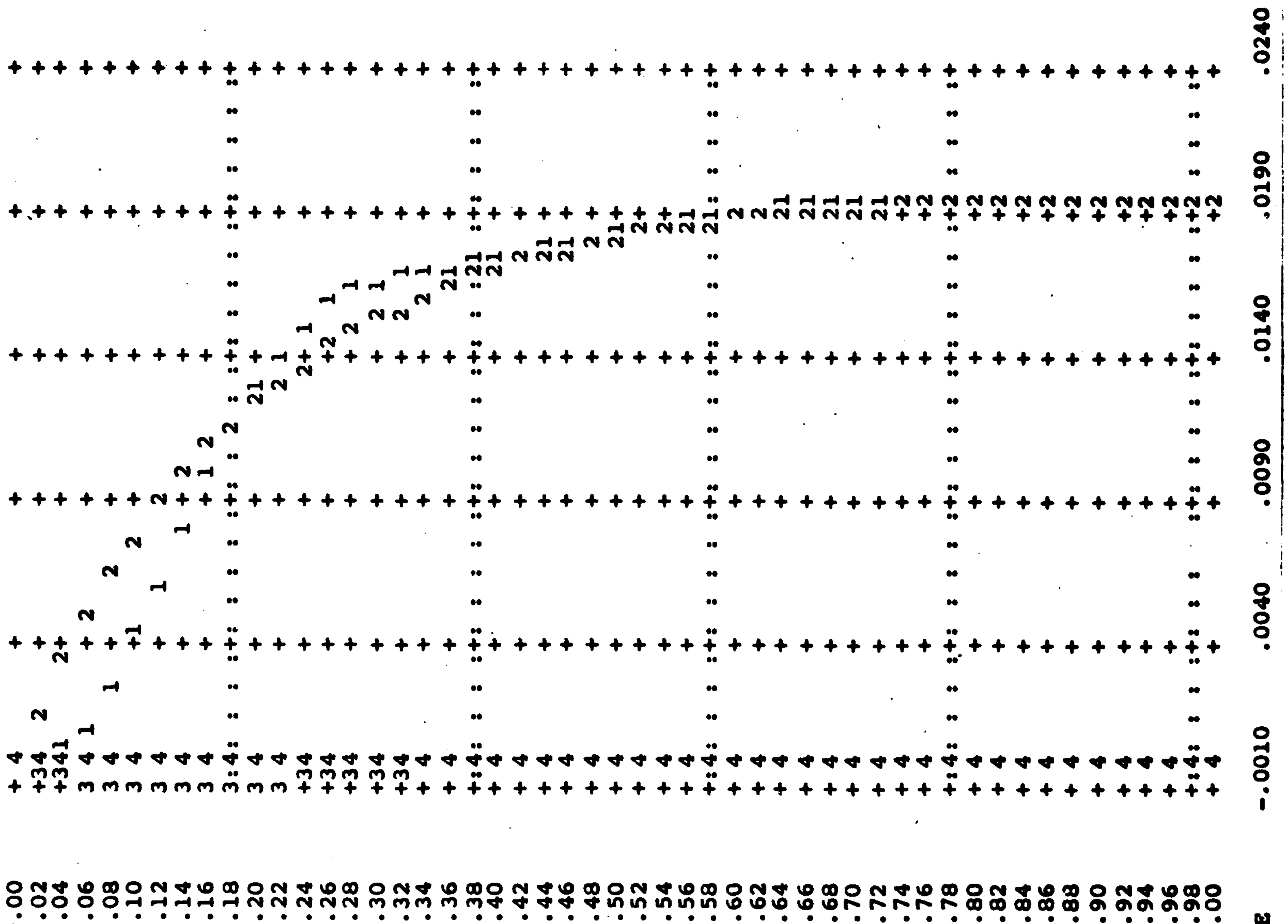

Fig. 6-21. AFTI(S3,A2,G3) Pitch-Pointing CGT/PI with Respect to Truth Model AFTI(S4,A2,G3)

The gains on system and actuator states in the matrix $\underline{K}_x$ and the gains of $\underline{K}_z$ and $\underline{K}_{x_m}$ are essentially identical to those of equations (6-60a,b) and (6-61a) determined for model AFTI(S3,A2). Also, the input gains $\underline{K}_{x_u}$ of equation (6-63b) above are identical to those of $\underline{K}_{x_u}$ in equation (6-61b) scaled by a factor of 0.02. Note that this is the same factor as between output to input ratios for the respective command models (see Section 6.4.5.1 above). Finally, the large gains on the angle-of-attack gust state (α_g) in the $\underline{K}_x$ matrix must be noted. These imply that the regulation of the α_g state will be very tight and accurate. Timely estimation of that state in the Kalman filter will be crucial to achieving desirable response to turbulence.

6.4.7 CGT/PI Pitch Rate Controller Design.

CGT/PI pitch rate controllers determined for design models AFTI(S3) and AFTI(S3,A2) are described briefly below. The effects due to actuator lags for this controller type are as noted previously (Section 6.4.6) for the pitch-pointing controller. Although the $\underline{\Pi}$ matrix of equation (3-47) proved to be rank deficient (non-invertible) because of the degenerate output matrix $\underline{C}$ defined in equation (6-50), the matrix pseudo-inverse was computed (automatically in CGTPIF) and the designs were successfully accomplished (Ref 32). An alternative would be to specify desired dynamics for a second output. However, for the pitch rate

AD-A115 510 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 1/2
DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS VIA COMMAND G--ETC(U)
DEC 81 R M FLOYD
UNCLASSIFIED AFIT/GE/EE/81-20-VOL-1 NL

3 OF 3
A115 510

END
DATE
FILMED
7-82
DTIC

controller, it is appropriate to specify only a desired pitch rate, allowing the natural aircraft dynamic relationships to determine the responses in the remaining outputs.

6.4.7.1 AFTI(S3) Pitch Rate CGT/PI. Quadratic weights providing desirable response qualities in the PI regulator and CGT/PI controller were readily determined. Weights of 150. on the output (output 1 only) and 50. on states θ and α, and of 1. on all control input magnitudes and rates (equal weighting for both inputs was arbitrarily selected and proved adequate) yielded PI feedback gains of

$$\underline{K}_x = \begin{bmatrix} -0.0357 & -0.4087 & -1.925 \\ -0.00909 & 0.4542 & -0.4875 \end{bmatrix} \tag{6-64a}$$

and

$$\underline{K}_z = \begin{bmatrix} -0.8266 & 0. \\ -0.2055 & 0. \end{bmatrix} \tag{6-64b}$$

Figures 6-22 and 6-23 show the CGT/PI output response and commanded control deflections, respectively. In Figure 6-22 plot symbols 1 and 2 are the actual and desired pitch rate responses; in Figure 6-23 plot symbols 1 and 2 are the horizontal tail and trailing-edge flap deflections. The pitch rate response achieves essentially perfect matching to the command model output response. The commanded pitch rate of 1° per second entailed maximum

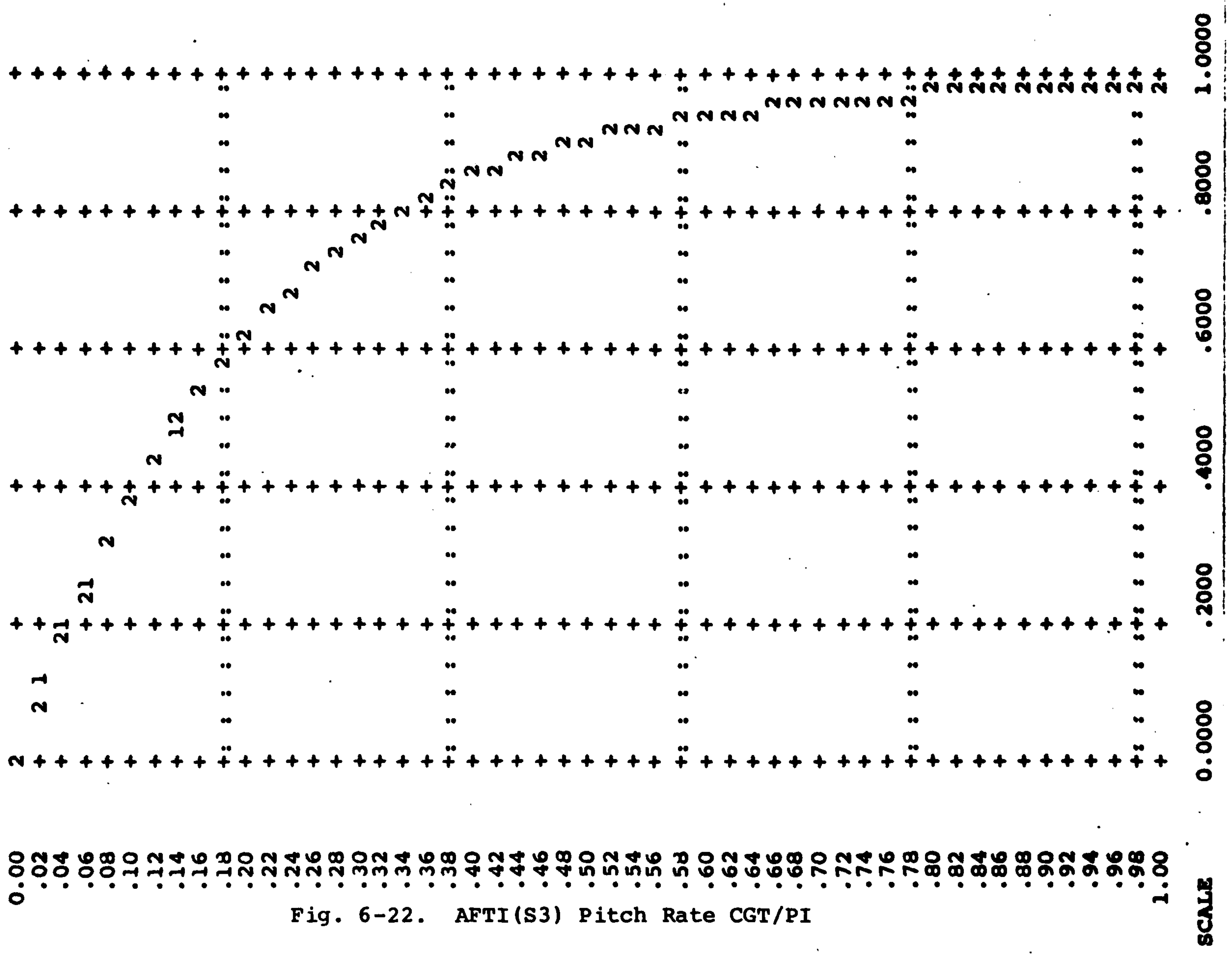

Fig. 6-22. AFTI(S3) Pitch Rate CGT/PI

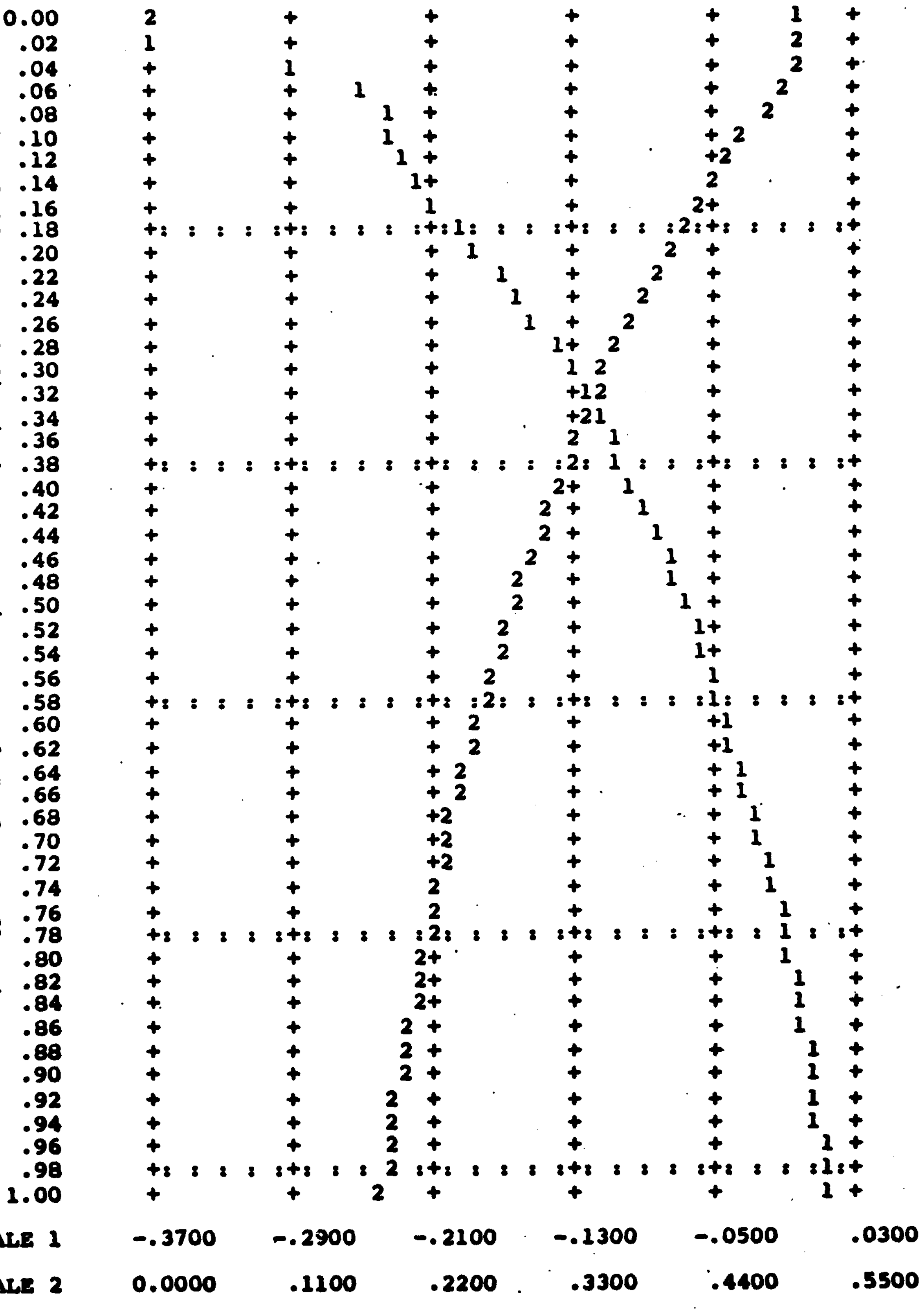

Fig. 6-23. AFTI(S3) Pitch Rate CGT/PI Control Surface Deflections

deflections of about -0.37° and +0.5° for the tail and flap, respectively.

The CGT/PI feedforward gains are

$$\underline{K}_{x_m} = \begin{bmatrix} -1.702 & 0. \\ -0.6385 & 0. \end{bmatrix} \qquad (6\text{-}65a)$$

and

$$\underline{K}_{x_u} = \begin{bmatrix} -0.3692 & 0. \\ 0.5074 & 0. \end{bmatrix} \qquad (6\text{-}65b)$$

Figures 6-24, 6-25, and 6-26 illustrate the performance of the CGT/PI pitch rate controller when subject to parameter errors in the design model. They show the response of the CGT/PI design described above when evaluated with respect to truth models $TM(S3)^{+}$, $TM(S3)^{-}$ and $TM(S3)'$. Response of the controlled system when subject to substantial errors (±20%) in the values of several key parameters yields response virtually indistinguishable from that for nominal design values. In the case of the response of Figure 6-26, even with gross errors in most of the design model parameters (due to a large change in flight conditions) the response is still usable despite some degradation.

6.4.7.2 <u>AFTI(S3,A2)</u> <u>Pitch</u> <u>Rate</u> <u>CGT/PI</u>. Using the same quadratic weights as discussed immediately above, the following PI regulator gains were computed:

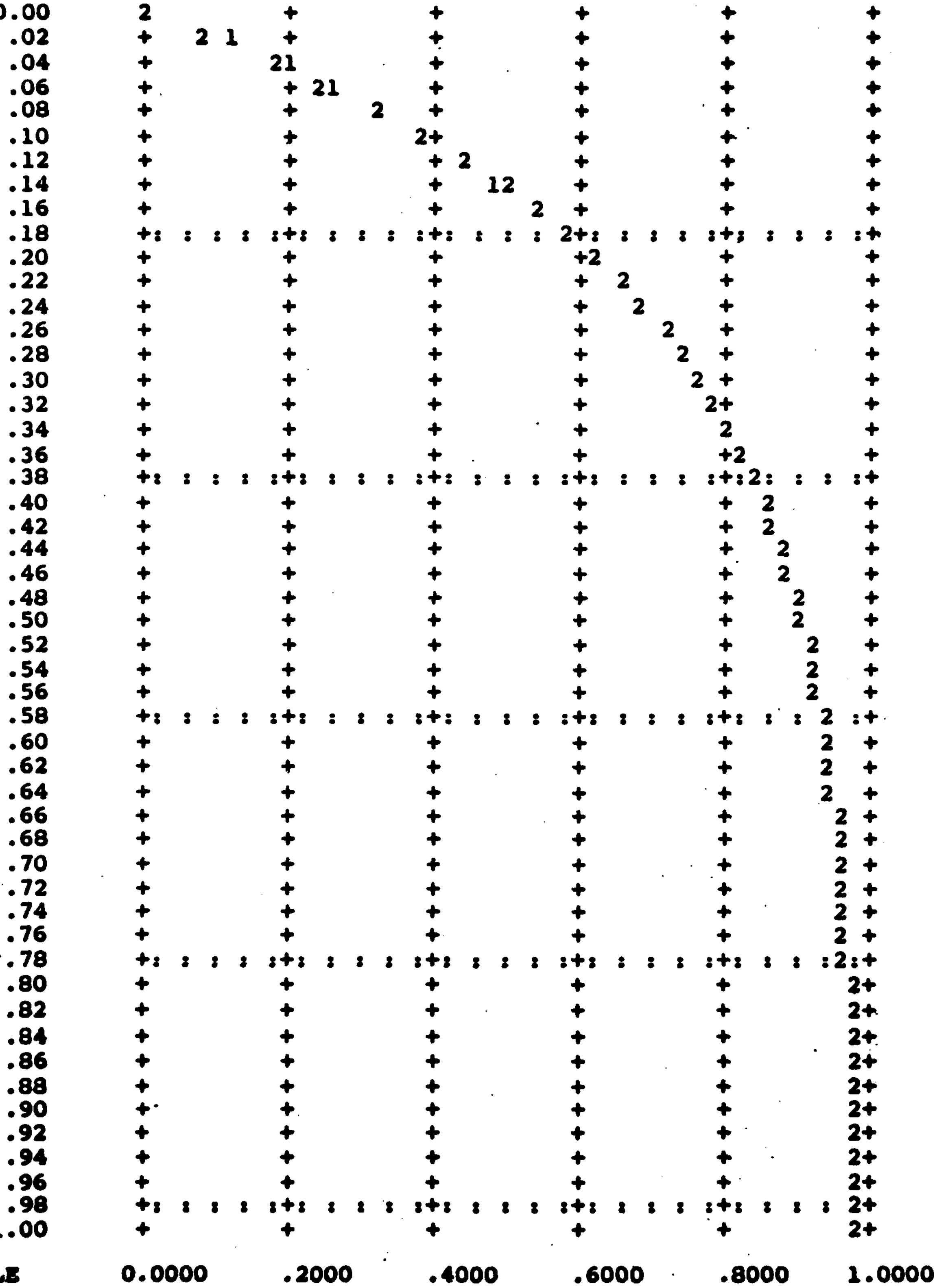

Fig. 6-24. AFTI(S3) Pitch Rate CGT/PI WRT TM(S3)

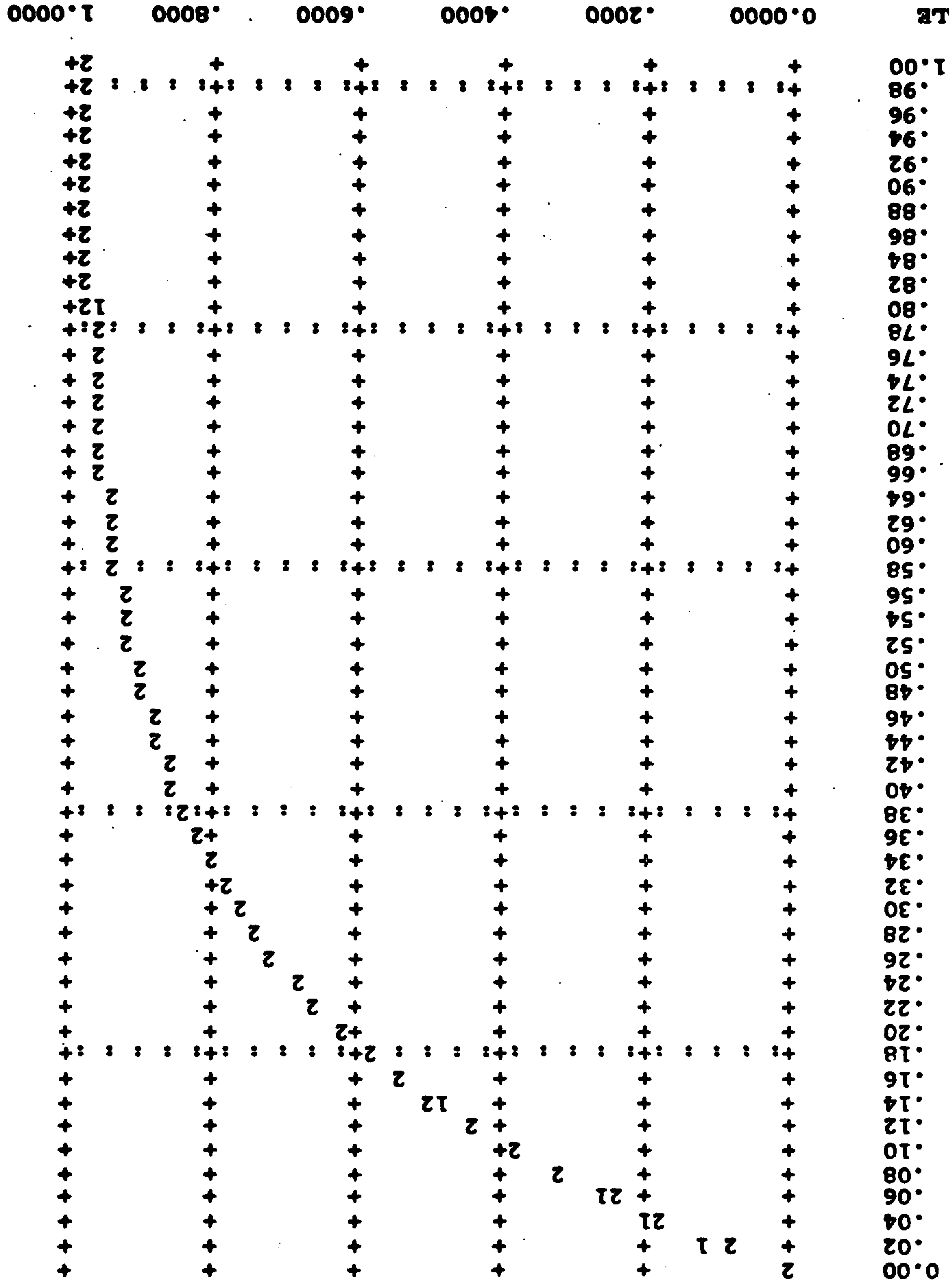

Fig. 6-25. AFTI(S3) Pitch Rate CGT/PI WRT TM(S3)

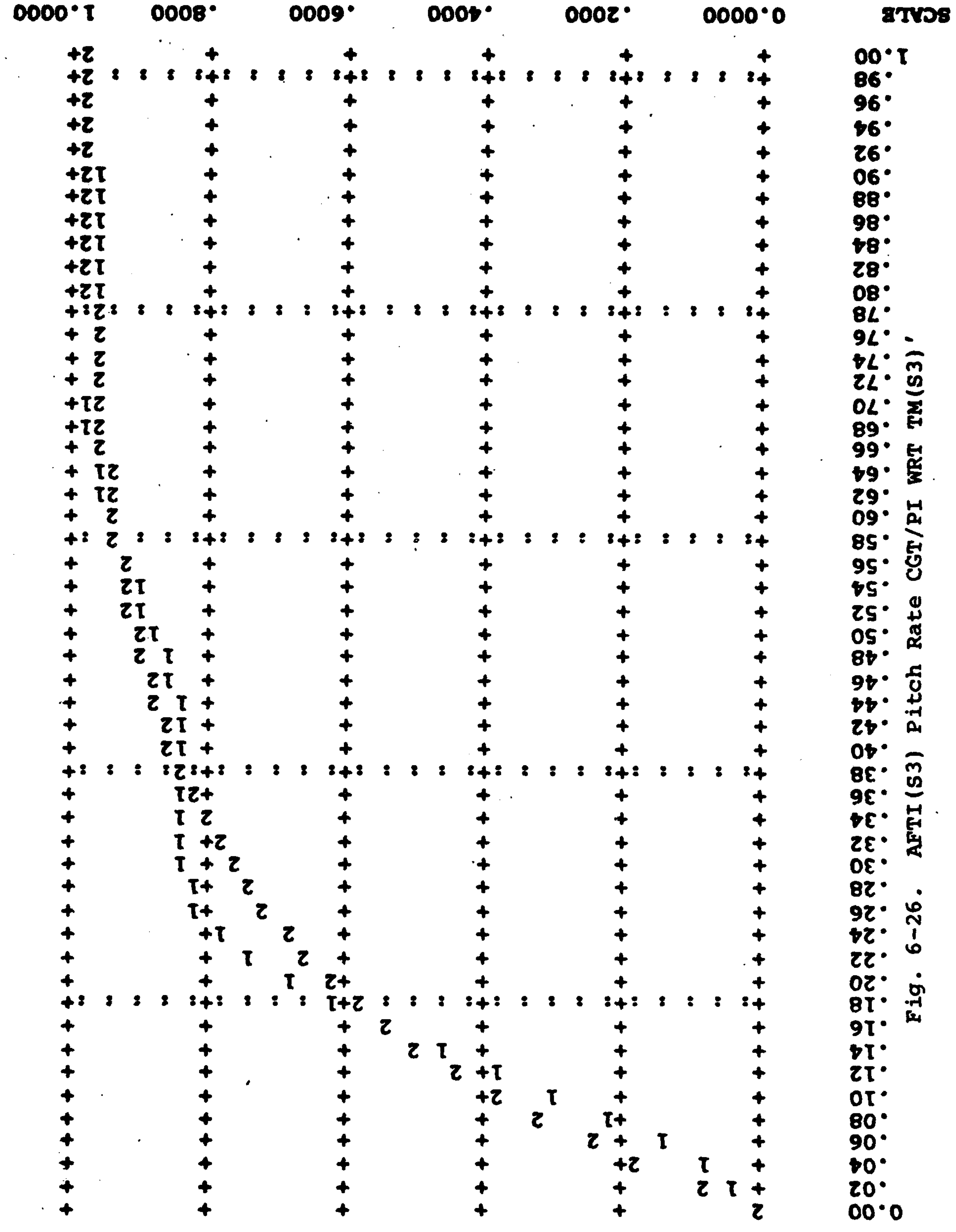

Fig. 6-26. AFTI(S3) Pitch Rate CGT/PI WRT TM(S3)'

$$\underline{K}_x = \left[\begin{array}{ccc|cc} -0.0349 & -1.357 & -2.313 & 1.376 & 0.1810 \\ -9.440\text{E}-3 & 0.9006 & -0.5936 & 0.1894 & 0.6637 \end{array}\right]$$

(6-66a)

and

$$\underline{K}_z = \begin{bmatrix} -0.6578 & 0. \\ -0.1803 & 0. \end{bmatrix} \qquad (6\text{-}66b)$$

The corresponding CGT/PI feedforward gains are

$$\underline{K}_{x_m} = \begin{bmatrix} -1.698 & 0. \\ -0.8371 & 0. \end{bmatrix} \qquad (6\text{-}67a)$$

and

$$\underline{K}_{x_u} = \begin{bmatrix} -1.151 & 0. \\ 0.8768 & 0. \end{bmatrix} \qquad (6\text{-}67b)$$

The response achieved by this CGT/PI controller is shown in Figure 6-27. Note that a short-lived and small-magnitude reversal in output response occurs at about mid-value in the rise portion of the transient response. Additional investigation of the state and control surface responses would be necessary in order to determine the cause of this observed reversal in pitch rate response.

The PI regulator was redesigned to overcome the observed reversal in output response. Increasing the output quadratic weight from 100. to 200. and the weights on θ and α from 50. to 100. gave much improved response. The alternate PI gains are

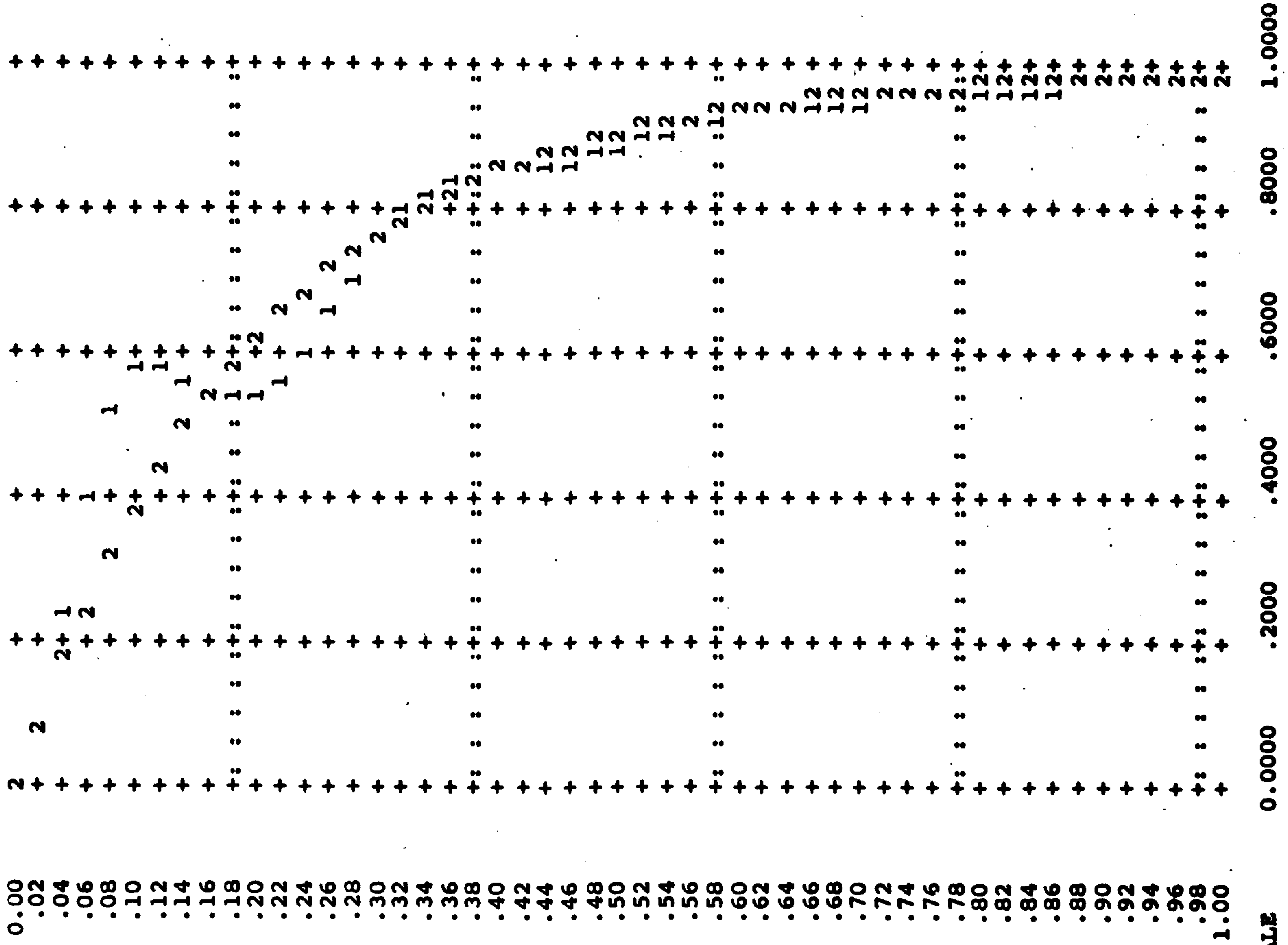

Fig. 2-27. AFTI(S3,A2) Pitch Rate CGT/PI

$$\underline{K}_x = \left[\begin{array}{ccc|cc} -0.0406 & -0.9833 & -2.595 & 1.430 & 0.1915 \\ -0.0122 & 2.657 & -0.6950 & 0.2031 & 0.6580 \end{array}\right] \tag{6-68a}$$

and

$$\underline{K}_z = \begin{bmatrix} -0.8027 & 0. \\ -0.2487 & 0. \end{bmatrix} \tag{6-68b}$$

The corresponding CGT/PI feedforward gains are

$$\underline{K}_{x_m} = \begin{bmatrix} -2.084 & 0. \\ -1.456 & 0. \end{bmatrix} \tag{6-69a}$$

and

$$\underline{K}_{x_u} = \begin{bmatrix} -0.8300 & 0. \\ 2.384 & 0. \end{bmatrix} \tag{6-69b}$$

The pitch rate response achieved by this alternate CGT/PI design is shown in Figure 6-28. Although the response is very good, this particular design would be unsatisfactory due to excessive control deflection. The flap is deflected a maximum of about 1.3° during an early transient for a pitch rate command of one degree per second. Element (2,1) of matrix $\underline{K}_{x_u}$ determines the initial command input to the flap actuator, so design iterations should be directed at achieving a smaller value for $\underline{K}_{x_u}(2,1)$ without increasing $\underline{K}_{x_u}(1,1)$ to the point that would cause difficulty with tail deflections (deflection limits for the horizontal tail are ±25.° while those for the

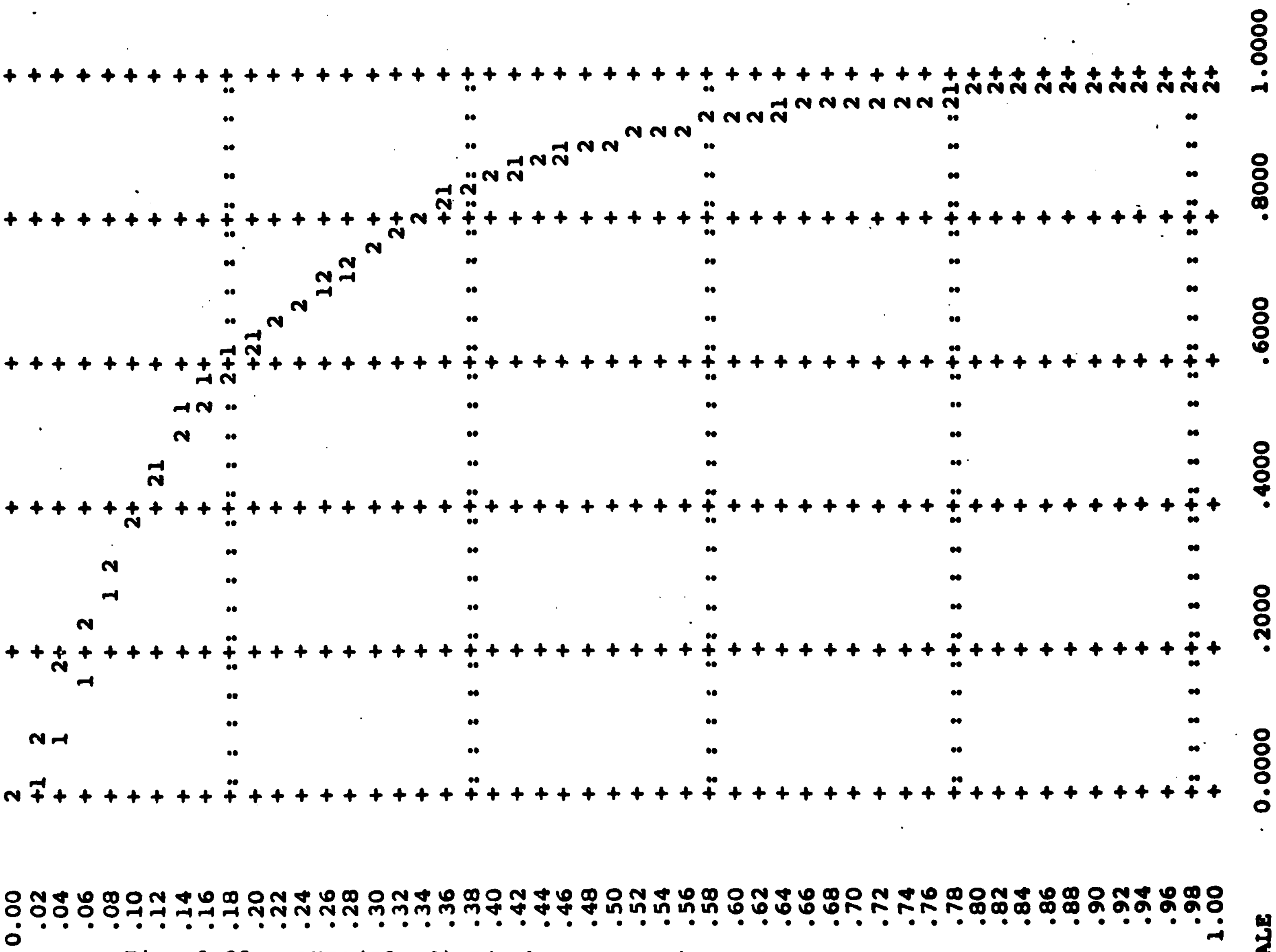

Fig. 6-28. AFTI(S3,A2) Pitch Rate CGT/PI--Alternate PI Regulator

trailing-edge flap are +20.° and -23.°). Different values of quadratic weights for q, α, and θ in the range of those used in the PI designs discussed above would undoubtedly achieve a satisfactory design.

6.4.8 <u>Kalman Filter Design</u>. A Kalman filter design for the AFTI/F-16 longitudinal flight controllers was readily achieved using the design model AFTI(S3,A2,G3) described above in Section 6.4.3.1 (based on measurements of θ, α, and q system states). The computed filter gain matrix is (see Equation 3-110),

$$\underline{\bar{K}}_F = \begin{bmatrix} 4.414E-2 & 1.620E3 & 9.911E-3 \\ 1.875E-2 & -2.698E-2 & -7.967E-3 \\ 6.704E-2 & -2.761E-3 & 4.576E-2 \\ 0. & 0. & 0. \\ 0. & 0. & 0. \\ -1.736 & 34.37 & 1.423 \\ -1.459E-2 & 0.2386 & 6.921E-3 \\ -3.477E-2 & 2.068 & -0.1628 \end{bmatrix} \qquad (6-70)$$

Note the two rows of zero elements are the filter measurement gains in updating the actuator states. This is due to the design model's representation of the actuator states as independent of driving noises and therefore having response that depends on the control inputs only with no uncertainty. Thus, as determined here, estimation of the actuator states would be based solely on the filter's internal dynamics model. This is clearly inappropriate since the actuator dynamics are not known perfectly and are actually subject to non-linearities (rate- and position-limiting), external dynamic forces (air loads affect achieved surface deflections according to flight condition), and atmospheric turbulence. Although it is desirable that flight controllers not include control surface deflection angles as feedbacks, it is clear from the earlier results of Sections 6.4.6 and 6.4.7 that the sampled-data implementations of the CGT/PI controller require such feedback. Therefore, measurements of actuator angles would be necessary and the design model should include artificial noise affecting the actuator dynamics as well as corruptive noise on the measurements, to yield nonzero elements where zeros exist in equation (6-70).

A covariance analysis was performed for this filter design with respect to truth model AFTI(S4,A2,G3) described in Section 6.4.2 above. The "true" and filter-computed estimation error covariances are computed for fifty filter sample periods by the method described in Section 4.4. The true and computed standard deviations for each state estimate during the fifty samples are output in plot form to the "LIST" file. In addition, the standard deviations at the final sample time are listed directly at the user's computer terminal. As discussed above, since the actuator states are modeled as determinstic, the estimation error is computed as zero for all time. Figures 6-29 through 6-34 are plots of the true and filter computed standard deviations for all design model states excepting the actuator states. Plot symbols 1 and 2 are the true and filter-computed standard deviations, respectively; the horizontal axis gives the time samples, and the dependent axes are scaled in units of radians or radians per second, as appropriate. The corresponding final-sample standard deviations (printed at the terminal) are (in units of radians or radians per second)

$$\begin{aligned} \text{For } \theta: \quad \text{true } \sigma_{\theta} &= 4.542\text{E-}4 \\ \text{computed } \sigma_{\theta} &= 4.727\text{E-}4 \end{aligned} \tag{6-71a}$$

$$\begin{aligned} \text{For } \alpha: \quad \text{true } \sigma_{\alpha} &= 4.994\text{E-}4 \\ \text{computed } \sigma_{\alpha} &= 5.313\text{E-}4 \end{aligned} \tag{6-71b}$$

$$\begin{aligned} \text{For } q: \quad \text{true } \sigma_{q} &= 1.236\text{E-}3 \\ \text{computed } \sigma_{q} &= 1.253\text{E-}3 \end{aligned} \tag{6-71c}$$

$$\begin{aligned} \text{For } \hat{\alpha}_g: \quad \text{true } \sigma_{\hat{\alpha}_g} &= 2.708\text{E-}1 \\ \text{computed } \sigma_{\hat{\alpha}_g} &= 3.044\text{E-}1 \end{aligned} \tag{6-71d}$$

$$\begin{aligned} \text{For } \alpha_g: \quad \text{true } \sigma_{\alpha_g} &= 1.865\text{E-}3 \\ \text{computed } \sigma_{\alpha_g} &= 2.095\text{E-}3 \end{aligned} \tag{6-71e}$$

$$\begin{aligned} \text{For } q_g: \quad \text{true } \sigma_{q_g} &= 2.067\text{E-}2 \\ \text{computed } \sigma_{q_g} &= 2.223\text{E-}2 \end{aligned} \tag{6-71f}$$

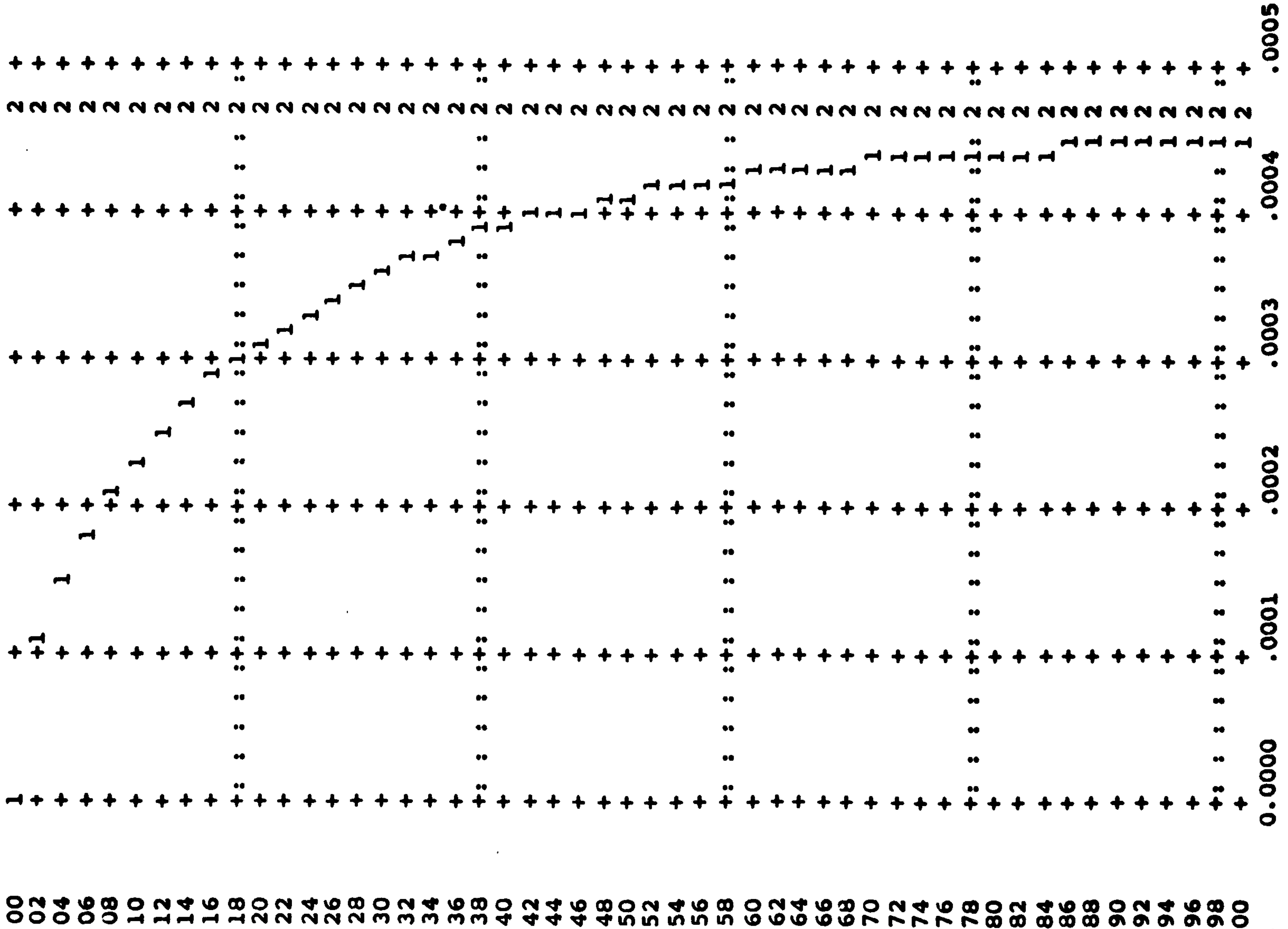

Fig. 6-29. True(1) and Computed(2) Standard Deviations of Estimation Error in θ

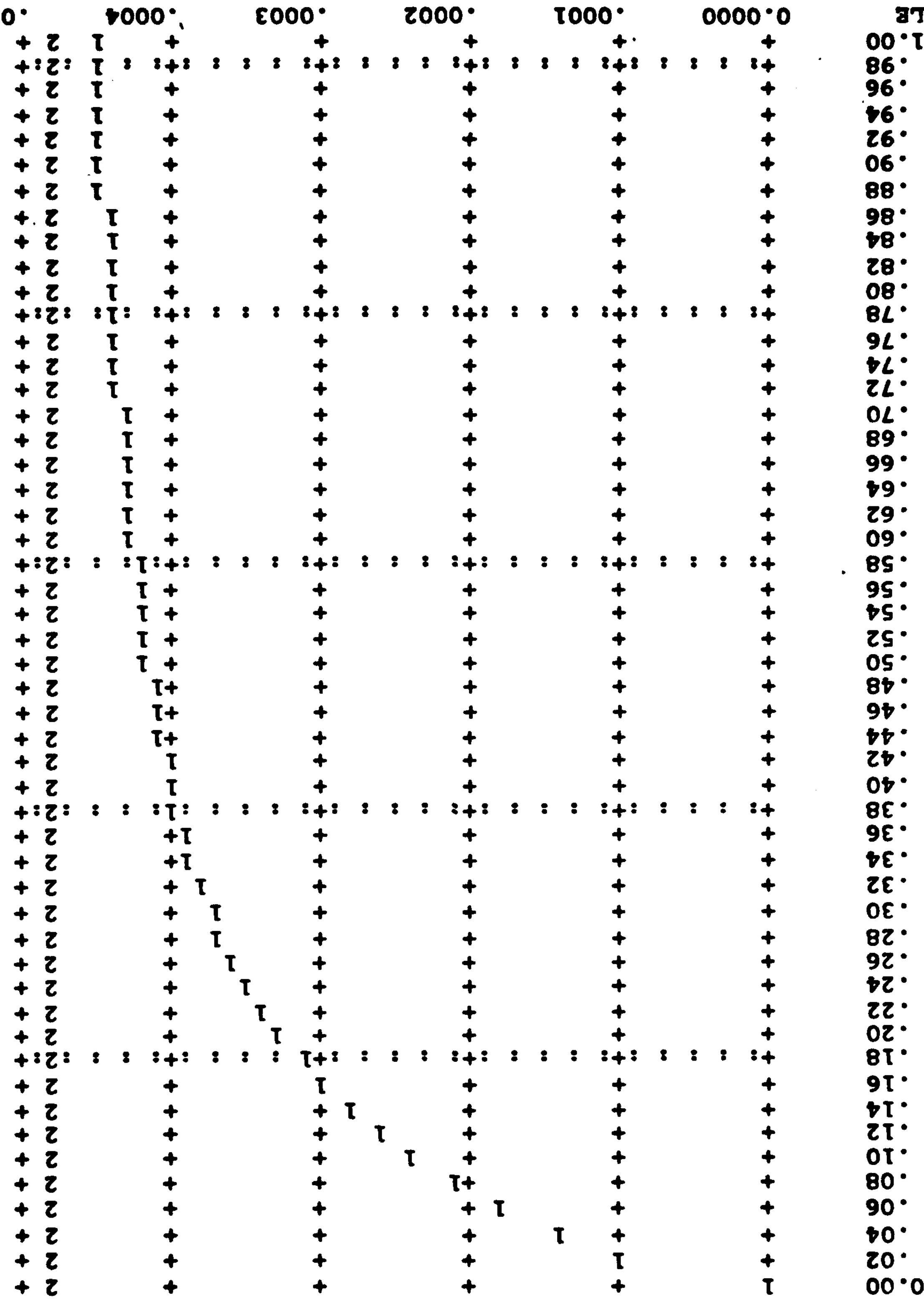

Fig. 6-30. True(1) and Computed(2) Standard Deviations of Estimation Error in α

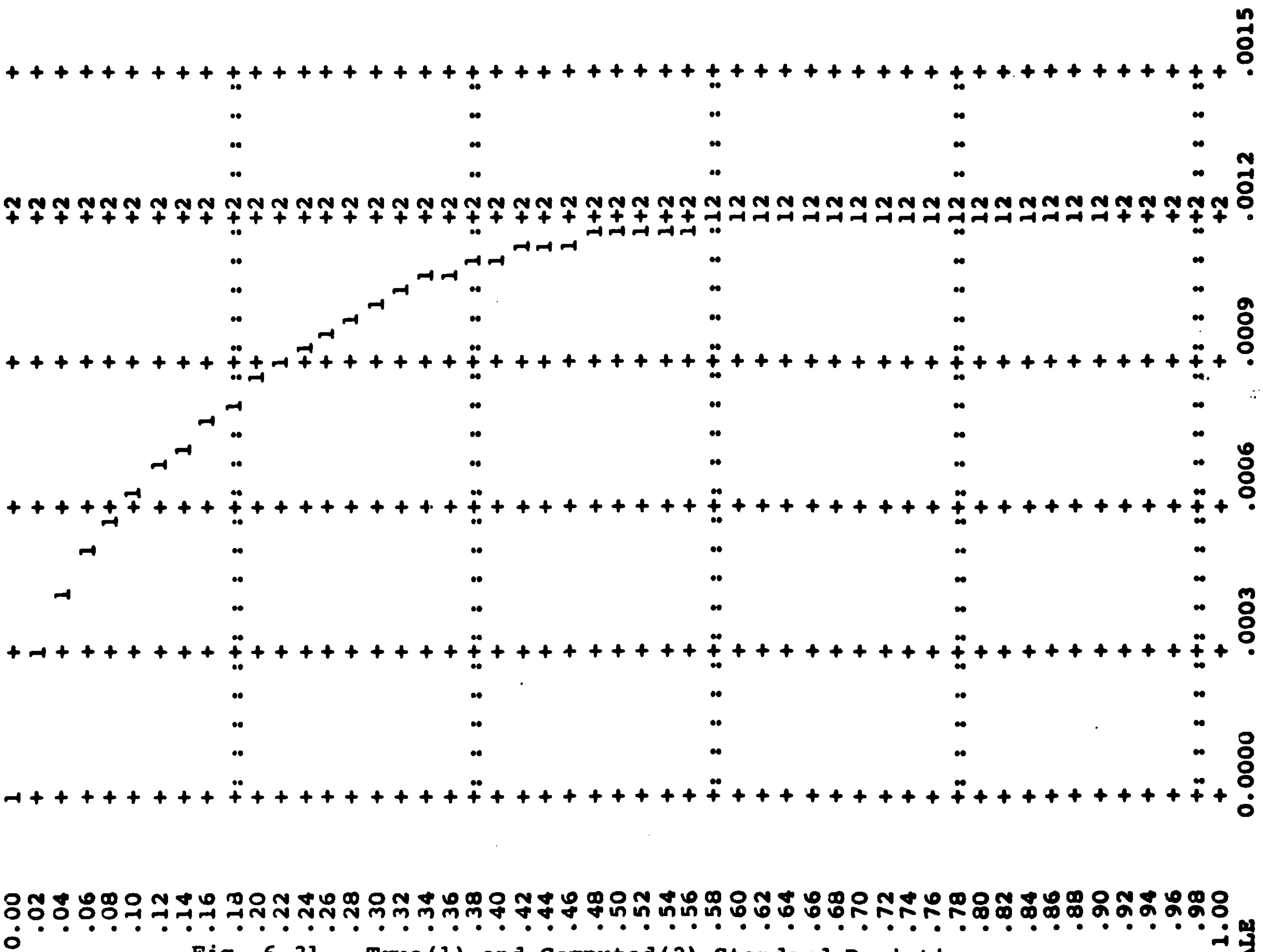

Fig. 6-31. True(1) and Computed(2) Standard Deviations of Estimation Error in q

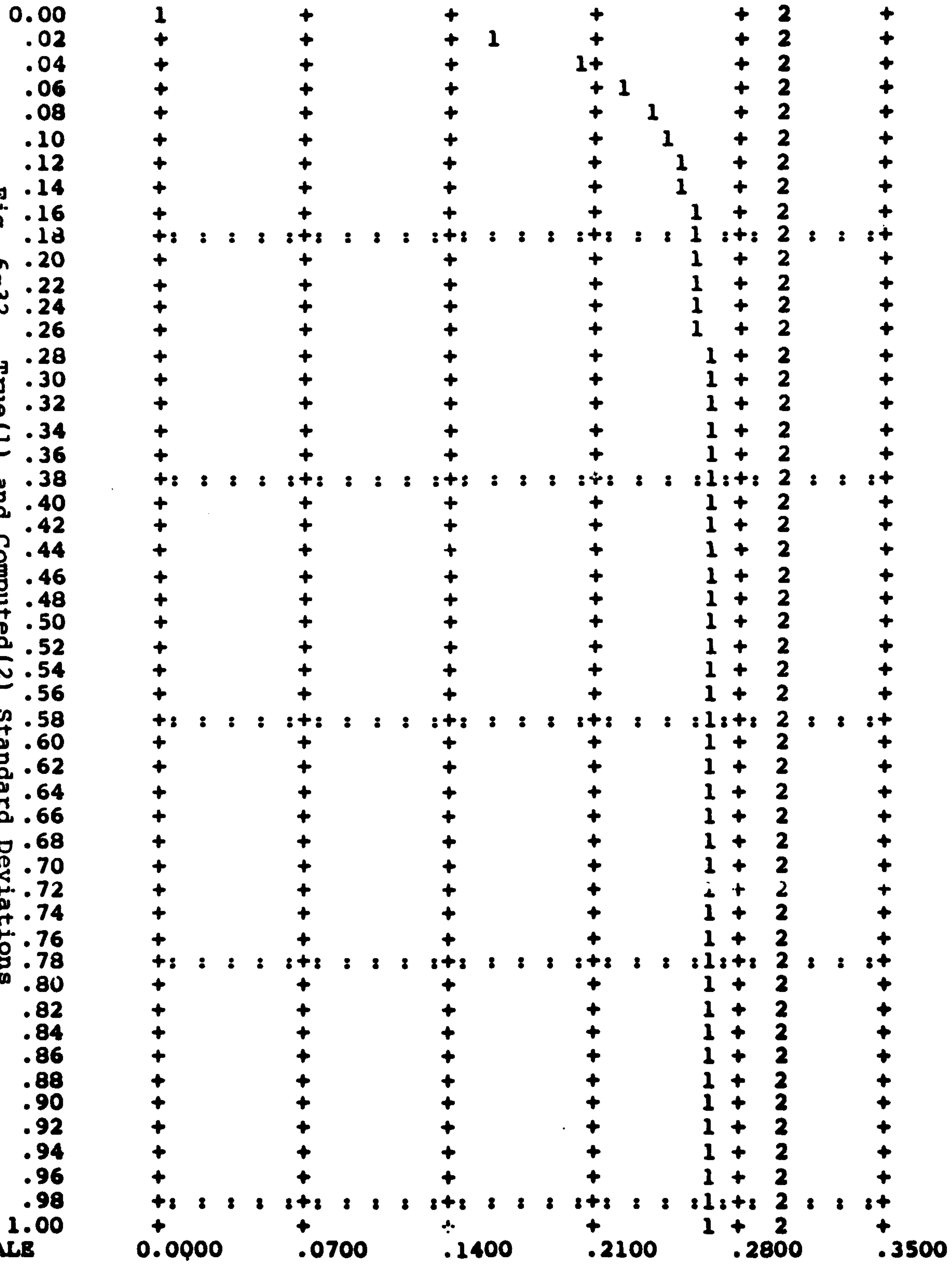

Fig. 6-32. True(1) and Computed(2) Standard Deviations of Estimation Error in α_g'

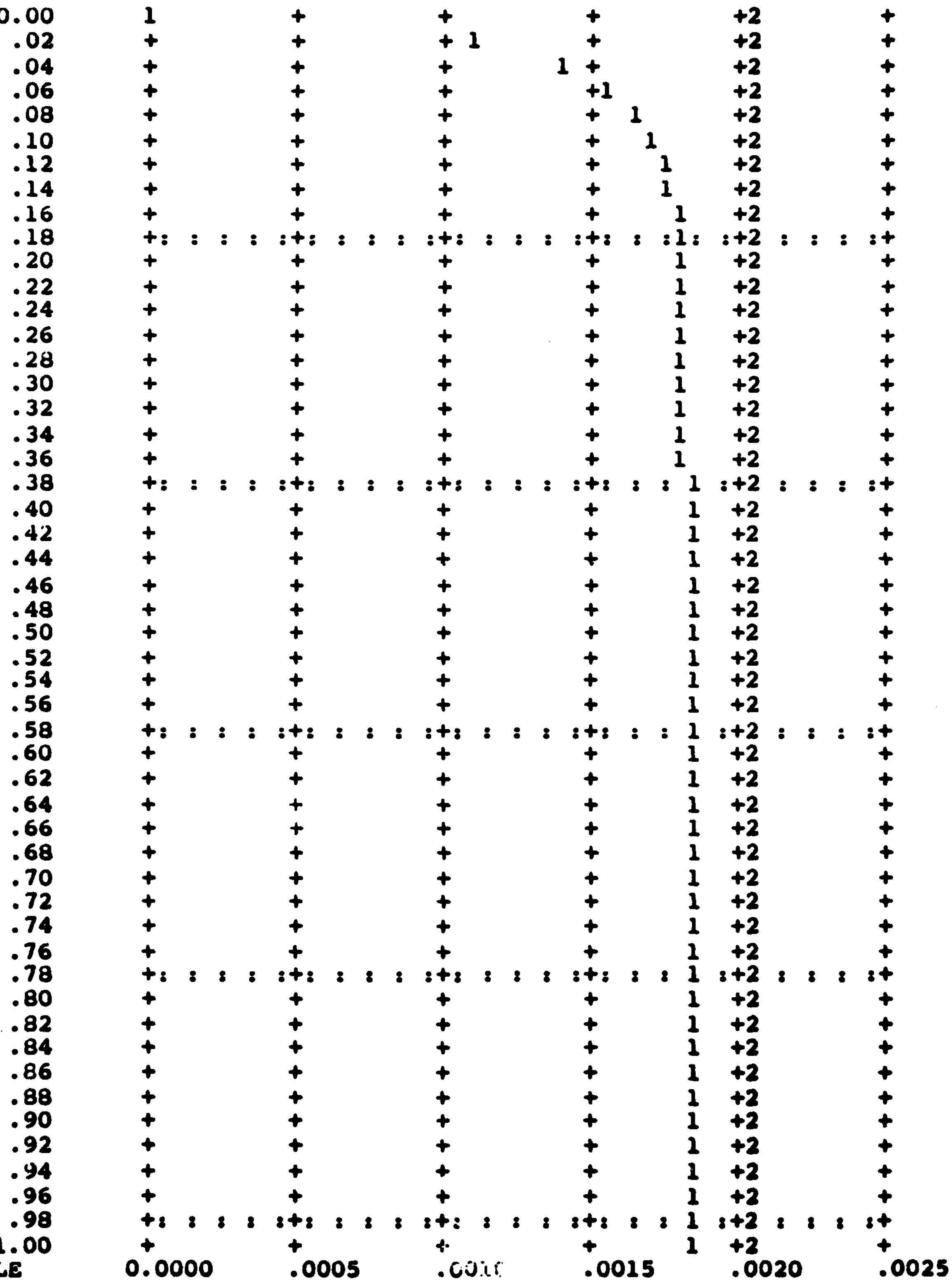

Fig. 6-33. True(1) and Computed(2) Standard Deviations of Estimation Error in α_g

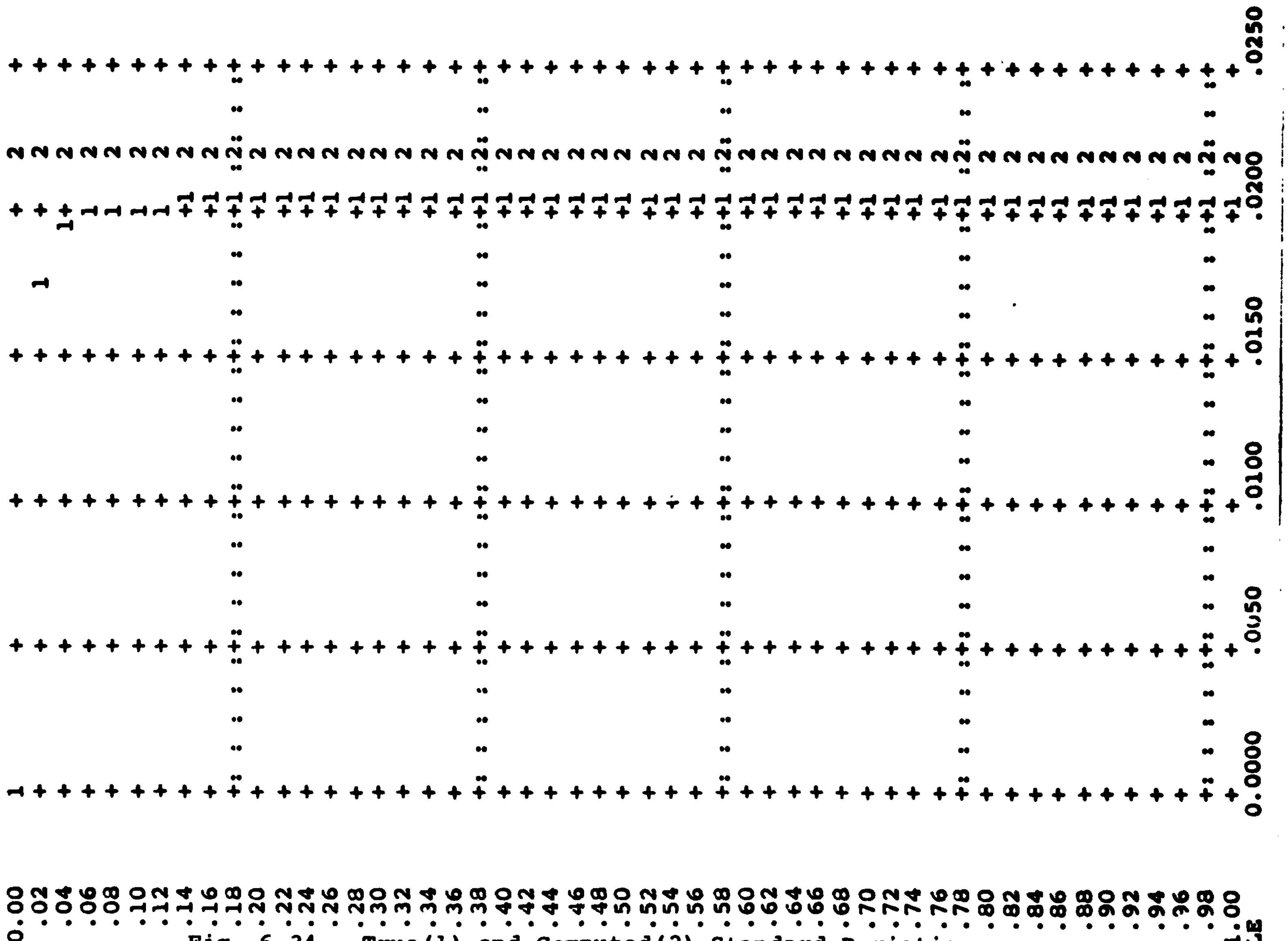

Fig. 6-34. True(1) and Computed(2) Standard Deviations of Estimation Error in q_g

As can be seen in equation (6-71d), the estimation error for the turbulence model state α_g' is quite large (about 16°) while that for state α_g is very good (about 0.1°). In the turbulence model developed in Reference 21, the angle-of-attack gust is modeled by a second-order system. In this representation (see equations (6-26a) and (6-26b)) the state α_g is the physical variable while state α_g' is simply a dummy state defined arbitrarily to achieve the desired second-order dynamics. Also, note from equation (6-62a) that the feedback gains on the α_g' state are smaller than those on the physical state α_g by a factor of about 0.003. Thus, the apparently large estimation error for α_g' is not a problem since it is a dummy state and the feedback gain on it in the controller is small. If desired, a transformation could be applied to the turbulence model so as to achieve comparable estimation errors for each of the two new angle-of-attack gust states.

These results show that the steady-state Kalman filter for this design problem achieves state estimation of adequate quality for the controller application. As mentioned above, this Kalman filter design assumed that control surface deflections are correctly modeled as deterministic. Successful controller/filter designs would need to model actuator states as non-deterministic and would probably require that measurements of the actuator deflection angles be taken, and made available to the Kalman filter. Finally, note that these results allow evaluation only of the filter alone. The combined controller/filter implemented as a CGT/PI/KF controller must be evaluated as an ensemble in order to judge the adequacy of the final and complete design. Additional design iteration of the CGT/PI and/or Kalman filter elements may be necessary in light of such an evaluation. Development of efficient software for evaluating an ensemble CGT/PI/KF is a strong recommendation for extending the work of this thesis. This would allow the controller to be evaluated in a realistic implementation and to explore the impact on system robustness in case Kalman filter estimates replace the assumed full-state feedback variables.

6.4.9 Discussion of Results for Flight Control Design Example. The designs discussed in this section have demonstrated the applicability of the CGT/PI and Kalman filter design and performance evaluation techniques to aircraft flight control problems. The CGT/PI designs achieve both conventional and decoupled pitch control. Command models providing the desired closed-loop dynamics are readily formulated. The controllers achieve good model-following and, in the case of perfect state knowledge, continue to give good model-following performance even in the face of parameter error in the design model dynamics description. Finally, the sampled-data implementation of the CGT/PI controller must be based upon a design model which includes models of control actuator lags in order to ensure stability when actuators with significant lags actually effect control surface deflections.

The Kalman filter design for the aircraft longitudinal dynamics subject to clear-air turbulence is readily achieved. The estimation errors determined from a covariance analysis prove to be sufficiently small to be used as feedback variables in the final closed-loop controller. Because of the possibility of large gains in the feedback and feedforward paths, the Kalman filter should be implemented so as to achieve minimum computational delay. Also it is noted that design models should include artificial noise sources driving the actuator states, and that measurements of the actuator deflections would be needed.

However, the evaluations described above provide insight into the CGT/PI controller and Kalman filter performances only as separate entities. In particular, the CGT/PI evaluation assumes that all feedforward and feedback quantities are known perfectly and with no delay. The final CGT/PI/KF controller requires state estimates from the Kalman filter in order to generate the control input commands. In this case, the filter would itself introduce additional dynamics and the state estimates are available only after a short delay. A complete evaluation of the full CGT/PI/KF controller requires additional software implementing the controller/filter system evaluation computations as described in Reference 32.

VII. Conclusions and Recommendations for Future Research

7.1 Conclusions

The objectives of this study have been twofold: (1) develop a user-oriented computer program aiding in the design of CGT/PI/KF controllers and applicable to a wide variety of specific control design problems, and (2) employ the design computer program to develop control designs relevant to aircraft flight control in order to evaluate the design process and qualities of the designs achieved. These objectives have been accomplished and lead to the following conclusions:

1. The design computer program (CGTPIF) developed during this study provides an effective means to pursue design of CGT/PI/KF controllers.

2. Command models in state variable form may be readily formulated for desired dynamic behavior described in terms of transfer functions (e.g., direct translation into standard controllable form). The command models may specify either conventional or decoupled input-output behavior for the closed-loop system. The model dynamics are expressed in the continuous-time domain and discretized within the computer program to determine the appropriate sampled-data controller.

3. CGT/PI designs achieving good model-following are readily achieved in a systematic, iterative design procedure. The PI regulator is first designed independently to provide rapid and well-damped output response to initial conditions. This design is then incorporated into the closed-loop CGT/PI control law and evaluated in terms of the model-following accuracy and control input magnitudes and rates for a step command input. Further tuning of the PI design is done iteratively (if necessary) to achieve the desired CGT/PI performance.

4. The CGT/PI/KF controller provides a design technique ideally suited to achieving controllers for sampled-data applications. The controller is a direct digital design, not employing approximations to transform a continuous design to discrete implementation.

5. The CGT/PI controller can be applied successfully to realistic flight control design problems. The design technique is particularly useful in achieving controller designs in the case of multiple independent control surfaces and multiple outputs. In such MIMO design problems, the CGT/PI controller provides all single- and cross-channel feedforward and feedback gains directly. By employing optimal control design methods based upon the Linear system-Quadratic cost weighting (LQ) assumptions, a stable closed-loop system is assured and the single and cross-channel gains are determined in a direct and logical manner.

6. In the case of perfect state knowledge, the CGT/PI controller maintains good model-following performance despite disturbances acting on system dynamics, and despite parameter errors of significant magnitude in the design model.

7. Design of the corresponding Kalman filter for the controller application is straightforward and presents no unique difficulties. For the flight control designs considered, a covariance analysis employing a system truth model demonstrated that state estimates of acceptable accuracy for use in the CGT/PI controller are obtained based upon a reduced design model. In actual implementation, actuator states would need to be measured and the design model should include pseudo-noise driving the control surface deflections in addition to the deterministic control actuator dynamics.

7.2 Recommendations for Future Research

As a result of experience obtained in designing CGT/PI/KF controllers in this study, several areas requiring additional study are apparent. The following recommendations include items directed at extending the CGT/PI/KF theory, modifying and extending the capabilities of the design computer program, and pursuing further evaluation of CGT/PI/KF designs.

1. Develop software extending the evaluation capability of the existing design computer program to encompass the complete closed-loop CGT/PI/KF controller. Such software is essential for evaluating the effects due to Kalman filter estimation of states required by the CGT/PI and will allow final tuning of the entire design. In particular, robustness of the CGT/PI and CGT/PI/KF designs can be conveniently compared with such software.

2. Modify the software in the PI regulator design path to allow quadratic weights to be applied directly to derivatives of the output variables; these would be manipulated by the program coding to achieve appropriate weights for states, inputs, and input rates. Improved damping of the output response can often be achieved by weighting both output and output rate deviations in the quadratic cost function of the optimal control formulation.

3. Develop (and implement) in the existing software) an extension to the CGT theory to provide for matching of the derivatives of the design and command model outputs (as well as meeting objectives as discussed in this research). This is analogous to the technique employed in the implicit model-following method and will tend to improve dynamics matching during the transient phase of response.

4. Using the software developed according to Recommendation 1 above, evaluate the flight control designs determined in this study as complete CGT/PI/KF closed-loop

controllers. Evaluation should focus on the quality of the model-following achieved for the designs in cases of nominal truth model parameters and erroneous truth model parameters. These will demonstrate the influence of filter dynamics and computational delay on the closed-loop controller performance and system robustness degradation due to filter estimation errors caused by modeling errors and parameter variation. Robustness recovery techniques (as described in Reference 32) may be employed in the final system formulation, if considered necessary.

5. Using the software developed according to Recommendation 1 above, compare the CGT/PI/KF controller performance with respect to disturbance rejection for flight control designs based upon design models in which disturbances (gusts) are modeled separately or included among the system states. The former case entails a distinct dynamics model for the disturbance states and the resulting CGT/PI/KF controller would employ feedforward gains on the estimated disturbance states to achieve disturbance rejection (CGT disturbance rejection). The latter case, in which the disturbance states are included in the design model's state vector, would result in a CGT/PI/KF controller which would employ feedback gains on the estimated disturbance states to achieve disturbance rejection (PI disturbance rejection). In either case, the Kalman filter design is invariant.

Bibliography

1. Asseo, S. J. "Application of Optimal Control to Perfect Model Following," Journal of Aircraft, 7: 308-313 (1970).

2. Athans, M. "The Role and Use of the Stochastic Linear-Quadratic-Gaussian Problem in Control System Design," IEEE Trans. Automatic Control, AC-16:529-551 (1971).

3. Barfield, A. F. Aircraft Control Engineer. Unpublished AFTI/F-16 Linear Aerodynamic Data. Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio, October 1980.

4. Barraud, A. Y. "A Numerical Algorithm to Solve $A^TXA-X=Q$," IEEE Trans. Automatic Control, AC-22:883-885 (1977).

5. Berry, P. W., J. R. Broussard, and S. Gully. "Stability and Control Analysis of V/STOL Type B Aircraft," ONR-CR213-162-1F. Office of Naval Research, Arlington, Virginia, March 1979.

6. Berry, P. W., J. R. Broussard, and S. Gully. "Validation of High Angle-of-Attack Analysis Methods," ONR-CR215-237-3F. Office of Naval Research, Arlington, Virginia, September 1979.

7. Bristol, E. H. "Designing and Programming Control Algorithms for DDC Systems," Control Engineering, 24:24-26 (1977).

8. Broussard, J. R. "Command Generator Tracking," TASC TIM-612-3, The Analytic Sciences Corp., Reading, Massachusetts, March 1978.

9. Broussard, J. R., P. W. Berry, and R. F. Stengel. "Modern Digital Flight Control System Design for VTOL Aircraft," NASA CR-159019. National Aeronautics and Space Administration, Hampton, Virginia, March 1979.

10. Broussard, J. R., Engineer. Personal correspondence. Information & Control Systems, Inc., Hampton, Virginia, May 4, 1981.

11. Cadzow, J. A., and H. R. Martens. Discrete-Time and Computer Control Systems. Englewood Cliffs, New Jersey: Prentice-Hall, 1970.

12. Chalk, C. R. et al. "Background Information and User Guide for MIL-F-8785B(ASG) Entitled Military Specification--Flying Qualities of Piloted Airplanes," AFFDL TR 69-72, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio, August 1969.

13. CYBER Loader Version 1 Reference Manual. Publication number 60429800. Control Data Corporation, Sunnyvale, California, 1979.

14. Davison, E. J., and S. H. Wang. "Properties and Calculation of Transmission Zeros of Linear Multivariable Systems," Automatica, 10:643-658 (1974).

15. Dorato, P., and A. H. Levis. "Optimal Linear Regulators: The Discrete-Time Case," IEEE Trans. Automatic Control, AC-16:613-620 (1971).

16. Elliott, J. R. "NASA's Advanced Control Law Program for the F-8 Digital Fly-By-Wire Aircraft," IEEE Trans. Automatic Control, AC-22:753-757 (1977).

17. Erzberger, H. "On the Use of Algebraic Methods in the Analysis and Design of Model-Following Control Systems," NASA TN-D-4663. National Aeronautics and Space Administration, Washington, D.C., July 1968.

18. Floyd, R. M. Aircraft Control Engineer. Unpublished AFTI/F-16 Aerodynamic Stability Derivative Data. Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio, June 1980.

19. Fortmann, T. E., and K. L. Hitz. An Introduction to Linear Control Systems. New York: Marcel Dekker, Inc., 1977.

20. Gran, R., H. Berman, and M. Rossi. "Optimal Digital Flight Control for Advanced Fighter Aircraft," Journal of Aircraft, 14:32-37 (1977).

21. Heath, R. L. "State Variable Model of Wind Gusts," AFFDL/FGC-TM-72-12, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio, July 1972.

22. Kalman, R. E., T. S. Englar, and R. S. Bucy. Fundamental Study of Adaptive Control Systems, ASD-TR-61-77, Wright-Patterson AFB, Ohio, 1961.

23. Kalman, R. E., and T. S. Englar. "User's Manual for the Automatic Synthesis Program," NASA CR-475. National Aeronautics and Space Administration, Washington, D.C., June 1966.

24. Kleinman, D. L. "A Description of Computer Programs Useful in Linear Systems Studies," Tec. Rep. TR-75-4. University of Connecticut, Storrs, Connecticut, October 1975.

25. Kreindler, E. "On the Linear Optimal Servo Problem," International Journal of Control, 9:465-472 (1969).

26. Kreindler, E., and D. Rothschild. "Model-Following in Linear-Quadratic Optimization," AIAA Journal, 14:835-842 (1976).

27. Kriechbaum, G. K. L., and R. W. Stineman. "Design of Desirable Airplane Handling Qualities via Optimal Control," Journal of Aircraft, 9:365-369 (1972).

28. Kuo, B. C. Digital Control Systems. Champaign, Illinois: SRL Publishing Company, 1979.

29. Kwakernaak, H., and R. Sivan. Linear Optimal Control Systems. New York: Wiley, 1972.

30. McRuer, D., I. Ashkenas, and D. Graham. Aircraft Dynamics and Automatic Control. Princeton, New Jersey: Princeton University Press, 1973.

31. Maybeck, P. S. Stochastic Models, Estimation, and Control, Vol. 1. New York: Academic Press, 1979.

32. Maybeck, P. S. Stochastic Models, Estimation, and Control, Vol. 2. Unpublished manuscript. Air Force Institute of Technology, Wright-Patterson AFB, Ohio, 1981.

33. Stein, G., and A. H. Henke. "A Design Procedure and Handling-Quality Criteria for Lateral-Directional Flight Control Systems," AFFDL TR-70-152, Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, Ohio, May 1971.

34. Tyler, J. S., Jr. "The Characteristics of Model-Following Systems as Synthesized by Optimal Control," IEEE Trans. Automatic Control, AC-9:485-498 (1964).

35. Winsor, C. A., and R. J. Roy. "The Application of Specific Optimal Control to the Design of Desensitized Model Following Control Systems," IEEE Trans. Automatic Control, AC-15:326-333 (1970).

Vita

Richard Michael Floyd was born on 28 March 1950 in New London, Connecticut. He enlisted in the Air Force in August 1972 and was assigned to RAF Croughton, England as a microwave radio repairman. Returning to the United States in September 1974, he attended the University of Connecticut under the Airman Education and Commissioning Program. He graduated Summa Cum Laude in May 1976 with a degree of Bachelor of Science in Electrical Engineering. During his undergraduate education, he became a member of Eta Kappa Nu and Tau Beta Pi. He received his commission from Officer Training School in September 1976. From September 1976 until June 1980 he was assigned as a flight control engineer in the Flight Dynamics Laboratory of the Air Force Wright Aeronautical Laboratories at Wright-Patterson AFB. He entered the School of Engineering of the Air Force Institute of Technology in June 1980 and has pursued the Guidance and Control curriculum of the Electrical Engineering Department.

Permanent Address: 59 Raymond Street
New London, Connecticut 06320

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GE/EE/81-20	2. GOVT ACCESSION NO. AD-A115 510	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) DESIGN OF ADVANCED DIGITAL FLIGHT CONTROL SYSTEMS VIA COMMAND GENERATOR TRACKER (CGT) SYNTHESIS METHODS		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Richard M. Floyd		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Wright Aeronautical Laboratories Flight Dynamics Laboratory AFTI Advanced Development Program Office Wright-Patterson AFB, Ohio 45433		12. REPORT DATE December 1981 13. NUMBER OF PAGES (Vol I) 213; (Vol II) 241
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

15 APR 1982

APPROVED FOR PUBLIC RELEASE AFR 190-17.

Lynn E. Wolaver
Dean for Research and Professional Development
Air Force Institute of Technology (ATC)
Wright-Patterson AFB, OH 45433

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR 190-17

F~~REDRIC C. LYNCH, Major,~~ USAF
~~Director of Public Affairs~~

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

MODERN CONTROL THEORY
DIGITAL FLIGHT CONTROL SYSTEMS
FEEDFORWARD CONTROL THEORY
MODEL-FOLLOWING CONTROL
PROPORTIONAL-INTEGRAL CONTROL

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

This study develops a computer program for interactive execution to aid in the design of Command Generator Tracker control systems employing Proportional-plus-Integral inner-loop controllers and Kalman Filters for state estimation (CGT/PI/KF controllers). Design parameters are specified in the continuous-time domain and the computer program obtains the corresponding discrete-time parameters and determines a direct digital design for sampled-data

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

implementation. Designs are based upon the Linear system model, Quadratic cost, and Gaussian noise process (LQG) assumptions of optimal control theory.

The report discusses the theoretical background and applications of optimal model-following designs which preceded the CGT theory. A development of the CGT/PI/KF controller theory is presented, and performance evaluation tools for the controller design are discussed. Following a brief description of the computer program developed, results of applying it to example aircraft-related controller design problems are presented and discussed. Among the designs presented are controllers for conventional pitch rate and decoupled pitch-pointing control for an aircraft system model representative of modern aircraft longitudinal dynamics. The CGT/PI/KF controller is found to be a technique particularly well suited to the typical aircraft control design problem wherein a multi-input multi-output (MIMO) system is to have specified output response behavior to commanded inputs ("handling qualities") while simultaneously rejecting disturbances of specifiable characteristics.

The computer program is fully documented in the appendices of the report. Included are a "Programmer's Manual," a "User's Manual," sample program input and output, a program listing, and a listing of job control language required to obtain an executable object file. These pertain to the computer program as implemented on a Control Data Corporation CYBER computer system and interactive execution under INTERCOM.